Spring 5-1-2005

University Enrollment Management Software

Ryan Knutson
Dakota State University

University Enrollment Management Software

Ryan Knutson

A project submitted in partial fulfillment of

the requirements for the Master of Science in Information Systems

Dakota State University

2005

MSIS
PROJECT APPROVAL FORM

Student Name: Ryan Knutson

Expected Graduation Date: May 05

Master's Project Title: University Enrollment Management Software

Date Project Plan Approved: May 6 '05

Date Project Coordinator Notified and Grade Submitted: _____

Approvals/Signatures:

Student: Ryan Knutson Date: 5/6/05

Faculty supervisor: Mark Moran Date: 5/6/05

Committee member: [signature] Date: 5/6/05

Committee member: Zhaohui Zhang Date: 5/6/05

To Ensure Certification of Completion:
Student must bring or send the original to the Graduate Programs Office.
Copies on acid free paper go to the library with the reports for binding.

Original to Graduate Programs Office
Acid-free copies with written reports to library
Copies to: Project supervisor and committee and to MSIS Coordinator

Abstract

Getting students through the doors in higher education is a continuing challenge. Declining budgets and increased competition are forcing enrollment offices to turn to technology. Technology allows them to reach out and connect with students across the nation and world. Access to state wide data allows staff to design, build and implement a communications system that will allow the enrollment office to communicate to prospective students. Data is downloaded from a central office that populates a Microsoft Access database. This data is then combined with Macromedia's Coldfusion web server application product, in order to develop a system to effectively manage prospective students. The methods, constraints, implementation and system requirements of the Enrollment Management Software are discussed in this report. Included in this document are the actual working screenshots of a successful implementation. Statistical representations of how the university was able to connect with more students than in previous years, along with additional plans for future versions as the enrollment office more heavily relies on this communication channel.

Table of Contents

List of Tables

List of Figures

Chapter 1 Introduction

This project consists of making the communications process more efficient and profitable for the enrollment office at Southwest Minnesota State University. The project entails designing a database to merge with the Minnesota State Colleges and University (MnScu) institutions through approved methods of data collection, along with designing the base communications system. The system will allow admission counselors the ability to communicate and manage their particular territories more efficiently. This system will be designed and created according to the specific criteria presented to the project developer and the criteria established from meetings with the potential users of the system. All design and implementation criteria will be driven by the specific goals of creating and implementing the project to meet timelines. The working system will be in use for the enrollment office for the recruitment year Fall 2004 – Spring 2005. Additional revisions and enhancements to the project will be ongoing throughout the entire calendar year and subsequent changes throughout the life cycle of the software. The project will leverage existing technology and be established in order to merge with existing and future web development projects for the University.

1.1 Statement of the Problem

State supported funding of higher educational institutes is taking a back seat to the funding of K through 12 causing declining budgets in state universities (Pawlenty, 2005). With declining revenues, institutions are beginning to strive to become more efficient and more profitable. One of the first areas of focus is retention (Danahar, 2004). Keeping existing students translates to immediately increased profits. While retention rates are

being improved, staff and budget cuts are taking place. Southwest Minnesota State University has lost two counselor positions while a third and fourth are out temporarily. Over half of the usual counselor staff had been lost and the pressure to increase enrollment meant that there is an immediate need to innovate.

The call went out for a software solution to aid in the enrollment problem. Software vendors were contacted and meetings setup to discuss packaged software solutions. Several large companies were brought on campus to discuss enrollment options. Each company spoke about how software could replace physical counselors by reaching out in new ways. Enrollment services, working under a reduced budget, found the cost of those software solutions to be out of range after reviewing the final bids. While researching packaged solutions, it became apparent that no software solution provider could get access to the direct downloads of the MnScu data, thus making for a disjointed solution (Gerber, 2004). It was impossible for vendors to tie into this data warehouse since it was both written specifically for the institutions and security policies prevented direct access. The only authorized viewers of the data were the institutions themselves. With these two obstacles the University needed to find an alternative method to solving the technology problem.

Data collection has been ongoing for many years in the enrollment office, however attempts at analyzing the data on a large scale have been time consuming processes. The need for accurate reporting of trends and other statistics is becoming a vital component in the science of enrollment (Goral, 2003). A technology solution that holds all of the information in one spot, would allow for a mathematical and statistical analysis of these numbers. This would help to guide the decision making process with

respect to the years enrollment activities. SMSU needed a technology solution that would solve all of their business problems in one contained package that fit the criteria that existed in the MnScu environment.

1.2 Objectives

The project objectives for the Enrollment Management System is to design, build, implement and enhance a working communications system that delivers prospective students relevant information through the use of a web application. This project must be tied to the MnScu data warehouse.

The objective of this project is to focus on the core functions that can become a working system. This system will then go immediately into production while balancing appropriate design and workflow processes. The ability to build a core function and implement while designing additional features will be required due to the fact that this project will be built in a live use environment. Flexibility in design and implementation will create a successful project.

The design phase of development consists of many meetings deciding what the product will look like, how it will function, requirements and an initial role out of a core-operating example.

Chapter 2 System Architecture

2.1 Scope

The project scope is a detailed listing of all the components that make up the enrollment management system. The scope focuses on delivery of the project far beyond the database design due to the requirements from MnScu and the timeline of the admissions office enrollment season. The project will include the following:

- **Database design and restrictions:** The database will be designed to the specifications that allow connections and updates via ODBC to a replicated database from MnScu.
- **Database table Emt_Prospective_Students:** This table is designed to hold the key data elements for prospective students.
- **Database table EMT_Contacts:** This table is designed to store individual counselor contacts for each student.
- **Database table EMT_Email_stats:** This table is designed to store individual emails that each particular counselor has sent to a group of students.
- **Enrollment Management Main Page:** A counselor enters through a login page and arrives at a main page that allows them to navigate the enrollment management system.
- **Set and Build Query Main Page:** A user arrives at this page in order to either choose from a list of queries or to fill out the criteria that allows them to create their own query. The user defines their query based on selected criteria.
- **Query Processing and Type Selection Page:** The user enters this page upon using the "build your query" page. This page compiles the query variables into a

- list along with asking the user to choose whether it is to be an "And" query or an "Or" query.
- **Report and Create Email Page:** The user is presented with the results of any query, separating the valid from invalid email results. The user is then able to draft a message to the group of valid email users with the ability to include additional criteria such as tracked links. Tracked links allow the system to count number of clicks from students.
- **Results of Report:** The user is presented with basic information for each student listed in the query. The page holds information such as Name, Email Address, Home Address, Phone Number, ACT, GPA, High School, etc.
- **Export to Excel:** This page allows the user to see the list formatted in such a way they are able to save the page as an excel document. This option allows the user to bring the list into other databases and also merge the list with a word document mail merge in order to populate a list of people to send postal mailings to.
- **View Listing with Contact Codes:** This page will allow the user to export the list to a separated format.
- **Create Email Page:** The user uses this page to fill out the information they would like sent to the student in the html email. They also use this page to select the tracked links and any other criteria to include in the email such as personalization.

- **Process Email:** The user will receive confirmation that their email has been sent to the student on this page. Once the bulk email process has been sent the user is routed back to the enrollment management home page.
- **Email Statistics:** This page is the primary statistical page that will let a counselor manage their workload along with managing their territory and producing lists with the proper contact information.
- **Prospective Stats:** The prospective statistics page reports a majority of the standard statistical features that provides an overview of who is in the prospective student pool and how they compare to previous years.
- **Extended Stats:** The extended statistics page is where the user finds additional statistics on the prospective pool such as Ethnicity, Gender, GPA, ACT, and Rank.
- **Territory Management:** The territory management page allows the counselor to add or remove high schools from their particular territory. This page allows self-management as territories change.
- **Enrollment Report Homepage:** The administrator uses this page to get a compiled list of statistics that are vital for managing the enrollment process.
- **Comparison Charts:** The administrator uses the comparison charts page to compare the current year statistics against the previous year statistics to determine progress and identify trends.
- **Gender:** The administrator views the gender page to determine what the break down of male to female prospective students exist in the database.

- **Ethnicity:** The administrator views the ethnicity page to determine what the break down of the different ethnic backgrounds that exist in the database.
- **State:** The administrator views the state page to determine what the break down of the different geographical locations the students are from.
- **Major:** The administrator views the major page to determine where the interest level of the different majors lie within the prospective database.
- **G.P.A.:** The administrator views the G.P.A. page to determine what the average G.P.A. is of the incoming prospective students.

2.2 History and Research: Southwest Minnesota State University Enrollment

Southwest Minnesota State University faced a dilemma. As the smallest of the four-year universities in the Minnesota State Colleges and Universities system (MnSCU) it has struggled through a period of declining budgets. Declining budgets were not the only problem as a fire had ravaged the Student Center two years earlier. With declining budgets and a depressed retention rate SMSU needed to energize their enrollment procedures (Shearer, 2004).

In discussing the records of previous students, the enrollment director displayed an area of the map at which gained the most students. A majority of the students who attended this university were first time higher education students from within a 100-mile radius. Although these tended to be a very excellent crop of students, the numbers from this region were beginning to decline as other markets encroached and populations began decreasing.

Successful marketing to areas outside of the 100-mile circle was identified as a potential problem for the enrollment services (Shearer, 2004). The ability to reach out and connect with students that were out of driving range had been a problem for the school from early on. Declining budgets reduced the ability to mass mail documents to these students and limited number of counselors that were assigned to an already expansive territory (Shearer, 2004).

Vendor technology research began to see if a product from a vendor would satisfy the requirements listed above. Three vendors were contacted and presented their software at different times throughout the academic year. The vendors included EMT Hobsons, James Tower, and RightNow technologies. Each solution provided a slightly different approach to the problem.

EMT Hobson's

EMT Hobson's is a company that has global ties to marketing and publications. They developed an electronic communications portal that allows admission offices to communicate to prospective students named EMT Connect. They accomplish this by uploading data sent to them through various means including a mailed cd or other means and then they allow counselors access into their system in order to send out emails to students. The costs associated to EMT Hobson's includes a set amount of email that can be sent and also database costs to manage the flow of databases that are needed to update their system. The actual software costs include an initial payment of $80,000 with follow up costs of approximately $5,000 per year ("EMT Hobson's", 2004).

RightNow Technologies

RightNow is a vendor who sells a CRM (Customer Relationship Management) software package ("RightNow", 2005). Although touted as an all in one sale, marketing and customer service package. The product began its life as an advanced self-help search product and the marketing and sales components have been added on throughout its life cycles. Although the company never gave the university specific figures as to cost they have a contract with the MnScu central office, which lowers its normal price to any member institution.

James Tower – Admissions Genie

The vendor James Tower, although approached the university in the later stages of development, produces a product that manages prospective students ("James Tower", 2005). Through the use of a database driven website, the system allows counselors to log in and email out prospective students. It also allows the students to log in and track their own progress through the enrollment process. The ability for the student to log in and track their own progress is unique from the two other vendors. James Tower purchased this five-year-old software product from an Atlanta based software vendor. This venture is a new one for James Tower and their price structure was around $80,000.

Each software vendor that presented on campus all had unique solutions to solving the electronic communication gap in slightly different ways. These software vendors are able to justify a higher price point than traditional software due to the expected return on investment gained from a percentage increase in enrollment. In an environment of decreasing budgets (Pawlenty, 2005) the packaged solutions, although

attractive, was out of reach for the university. The ability to get and maintain data from the central data warehouse also became a support problem, as the vendors were not allowed direct connection to the data storage area. These negatives convinced Southwest Minnesota State University to seek in house design and build options in order to solve their problem with minimal capital outlay. The in house solution provided an environment where all available resources could be pooled for a cost savings solution that allowed a direct connection to the MnScu data warehouse.

Chapter 3 Design Aspects

3.1 Designing the Database

The enrollment project specifications required certain database elements to exist. In order for the project to be successful it required a properly designed database. To complete the prospective enrollment communication system, the proper data needed to be obtained from the central database server. The MnScu system is the central information technology and storage area for all of the universities data. This meant that the holder of all of SMSU's information also controlled access to and retrieval of that same data.

Consultation with MnScu personnel revealed that the only way to download and use the data of the prospective students was to link Microsoft Access database tables through the replicated database storage system (Thompson, 2004). The MnScu central office also did not allow a way to upload data to be stored. These two conditions meant strategic planning of database design would be needed to achieve a product that was both useful and complete.

An additional factor involved in designing the database was the coding language used by the school. SMSU uses ColdFusion from Macromedia (Now owned by Adobe) as its web application software ("Macromedia", 2005). This software allows you to connect a website to a backend database in real time. In order to achieve a system that allows admissions personnel to design and create their own queries, the database tables must be built in such a way that they place all relevant data in one master table with additional supporting tables. With these two limitations in mind, the database could be created to house all of the prospective students for the university.

3.2 Database Structure

Basic table structure of the database included merging a majority of the key component data into a single table named emt_prospective_students. This table holds the key information pertaining to the prospective student. By combining these elements into a single data table it allowed for easy access through the creation of dynamic query building. See Appendix C for database structure. Need to store multiple contact records for each student dictated that another table was created. This table would be a one to many relationship (relational) in that each student would have multiple contacts (Hoffer, & Prescott, & McFadden, 2002). The best way to store data in this fashion is to create a relational database. This is done by maintaining a common key and breaking the data out to a separate table. The table EMT_Contacts stores each contact code for all of the prospective students sorted into individual fields. See Appendix C for database structure.

The database also needed to store the information that was being sent to the students in the email. This included data elements such as the query used and number of students contacted. Without proper storage of these elements, enrollment administration would not be able to manage the flow of communications that admission counselors were sending. Therefore, EMT_EMAIL_STATS plays a key role in the basic reporting data structure.

The ability to manage data in a timely fashion becomes crucial when the admissions office is in a period of enrollment. With the admissions office using these screens daily to report progress of enrollment, the need to have them updated is vital. To achieve a daily update, a script was written that draws from a replicated database through an ODBC connection at the central office on a nightly basis. A batch file is a script that

executes in the command line of a Microsoft Windows computer. It allows you to automate tasks through the use of command line tools ("Webopedia", 2005).

Table 3.1: Batch File Commands

BATCH FILE CODE TO AUTOMATE UPDATE	
Copy h:\inetpub\wwwroot\cgi-bin\emt.mdb c:\temp	Used to copy the web server database to the clipboard
Copy c:\temp\emt.mdb c:\temp\temp	Copies database to safe location in case of error
Cd \Program Files\Microsoft Office2000\Office\ MSACCESS.EXE c:\temp\emt.mdb /x autorun_everything	Changes the command line location to the office folder, runs Microsoft Access, opens the emt.mdb database, and runs the autorun_everything macro
TASKKILL /F /IM MSACCESS.EXE	Windows XP command line tool to stop a process, Microsoft Access
Copy c:\temp\emt.mdb h:\inetpub\wwwroot\cgi-bin\	Copies updated database to the Web Server drive.

3.3 Application Components

The enrollment management system dictated the proper configuration of hardware in order to function properly. A solid database, web server, application server and client web browser was crucial in designing a system that would be flexible enough to access on campus or off. This particular challenge meant additional needs surrounded the security of the web application.

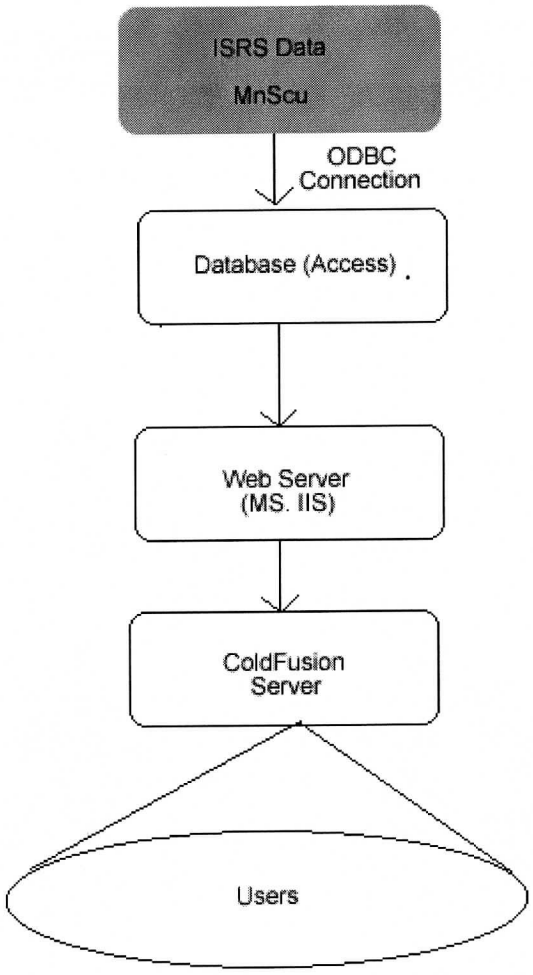


Figure 3.1 Application Paths

3.4 Security Framework

Security is a vital aspect of the project; a breach of security would expose all of the prospective student pool for the upcoming semesters. If a corruption or security breach were to occur the intruder would have the ability to email the entire database of prospective students and the university would have the potential to lose an entire crop of students for the upcoming academic year. Since counselors need to access the email system from the road during high school visits, the system must exist behind a secure log in. By using session variables during a log in process, the software ensures that only authorized users access each page. Server side authentication adds an additional layer of security to help protect the database. By authenticating on the server side the software is passing variables back to the application server. The application server then processes those variables and stores them on the application server. This means that they do no exist on the users machine. Implementing session variables by only existing on the server they are much harder to intercept and change by outside users.

3.5 Primary Application

The goal of the enrollment management software is to allow users to query a list of students and communicate to them through the use of email. The user logs into the system and is greeted with a homepage that allows them to decide what path they will take through the software. The security architecture defines what a user will see on the application homepage. If the user has administrative rights they will be able to select any option, however if logged in as a counselor, they will only be allowed to view those areas that the Enrollment director has allowed them to see. The counselors will have access to

areas such as the generating of lists and reports. The lower security level users would not be able to view the progress of the other counselors or edit the report forms. If the user selects to generate a list of students or send an email to that list, they would select the "Generate Lists/Emails".

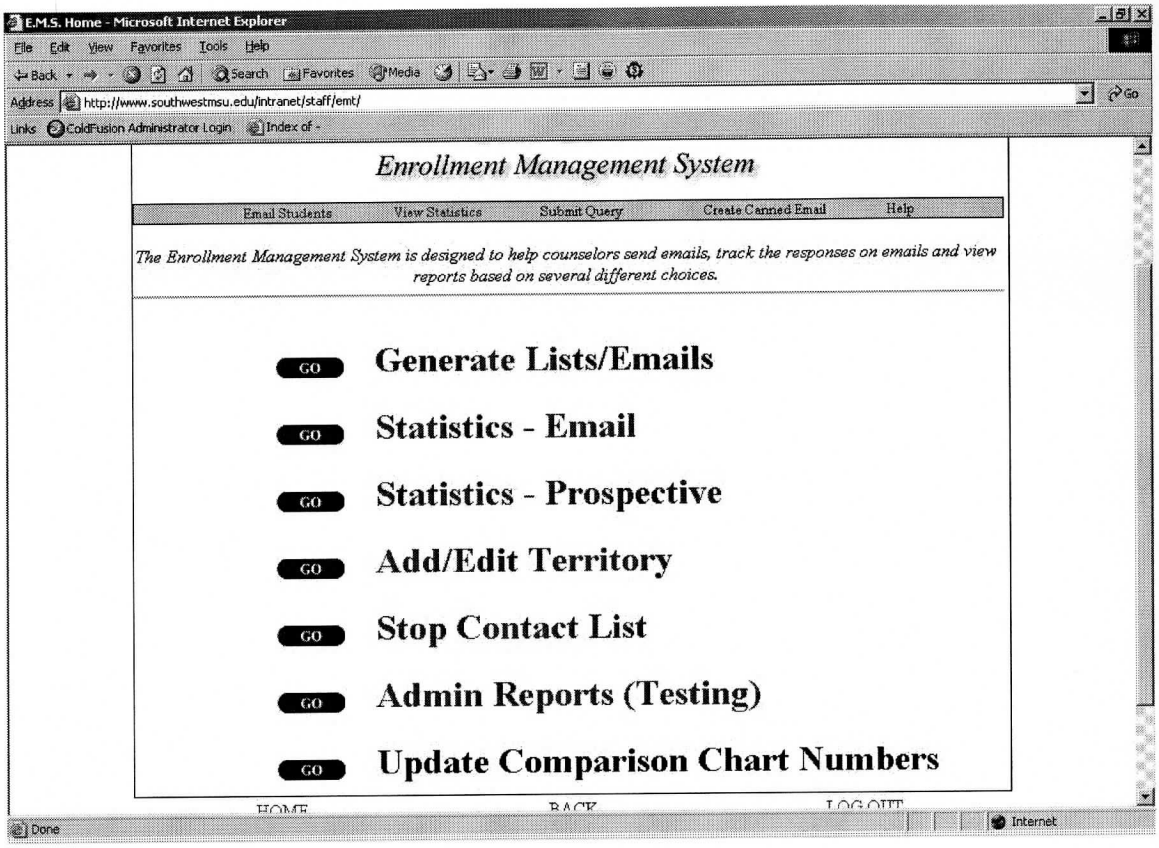


Figure 3.2 Enrollment Management Homepage

Once they have selected to generate a list or an email to students they would then need to define what criteria at which they would be querying out the students.

3.6 Dynamic Query Building

A crucial requirement for the Enrollment Management Software was to allow the counselors to create the queries dynamically from a series of dropdown boxes. By allowing the counselors to create their own dynamic queries they can pull student records based on their own criteria. This meant if they wanted to select students with an ACT over 20 and a class rank over 100 they could choose those options without consulting support staff. The initial planning stage included an interview with counselors to see what type of groupings of students would be useful in their communication process. As a result of the interview a list was created of potential queries ("Counselor Interview", 2004). The list of queries would need to be converting to SQL so that they could be included in the core of the query builder pages. Whereas other software solutions must train their users in writing SQL, it was important that the user be able to utilize the technology with very limited training ("EMT Hobson's", 2004).

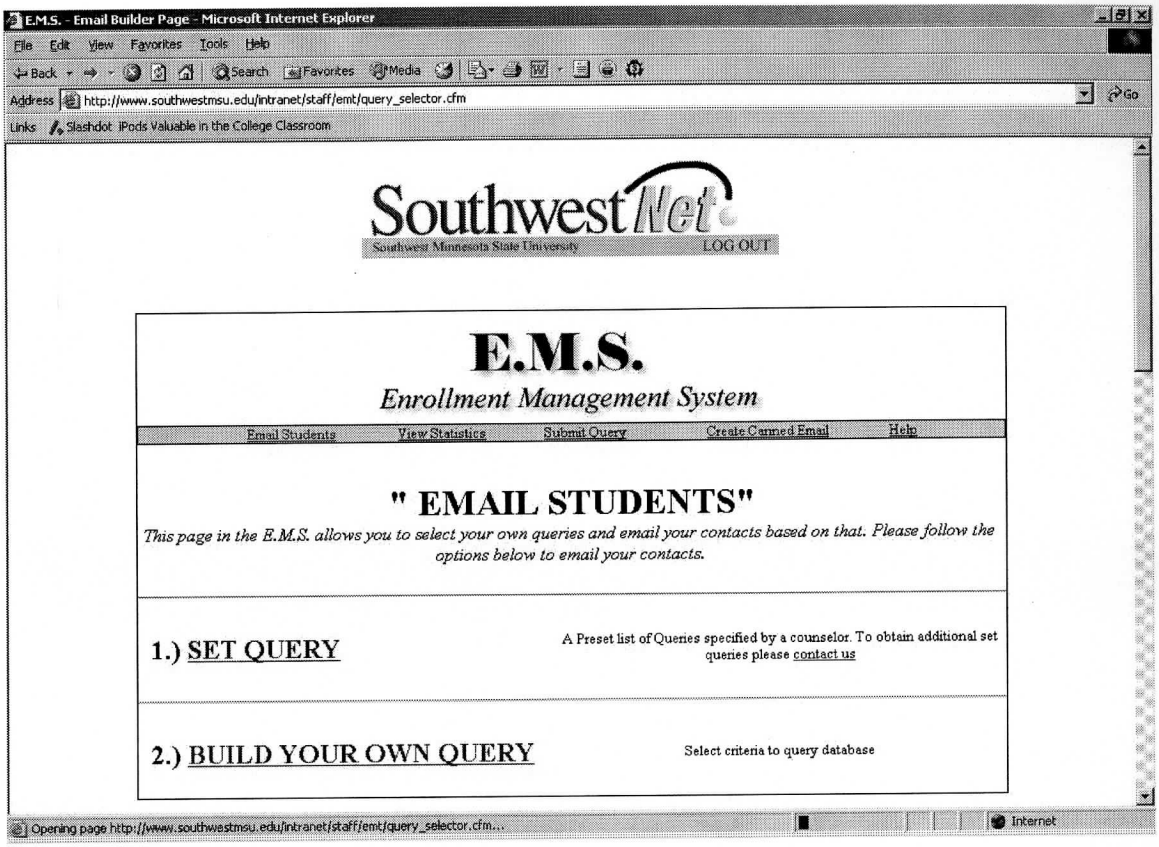


Figure 3.3 Set and Build Query Page

Set Query

The first option available is to choose from a list of set queries. These set queries consist of dropdown box of predefined queries built by support staff that a user can select and process. Support staff built these queries since they would not be able to accomplished through the use of the build your own query mechanism.

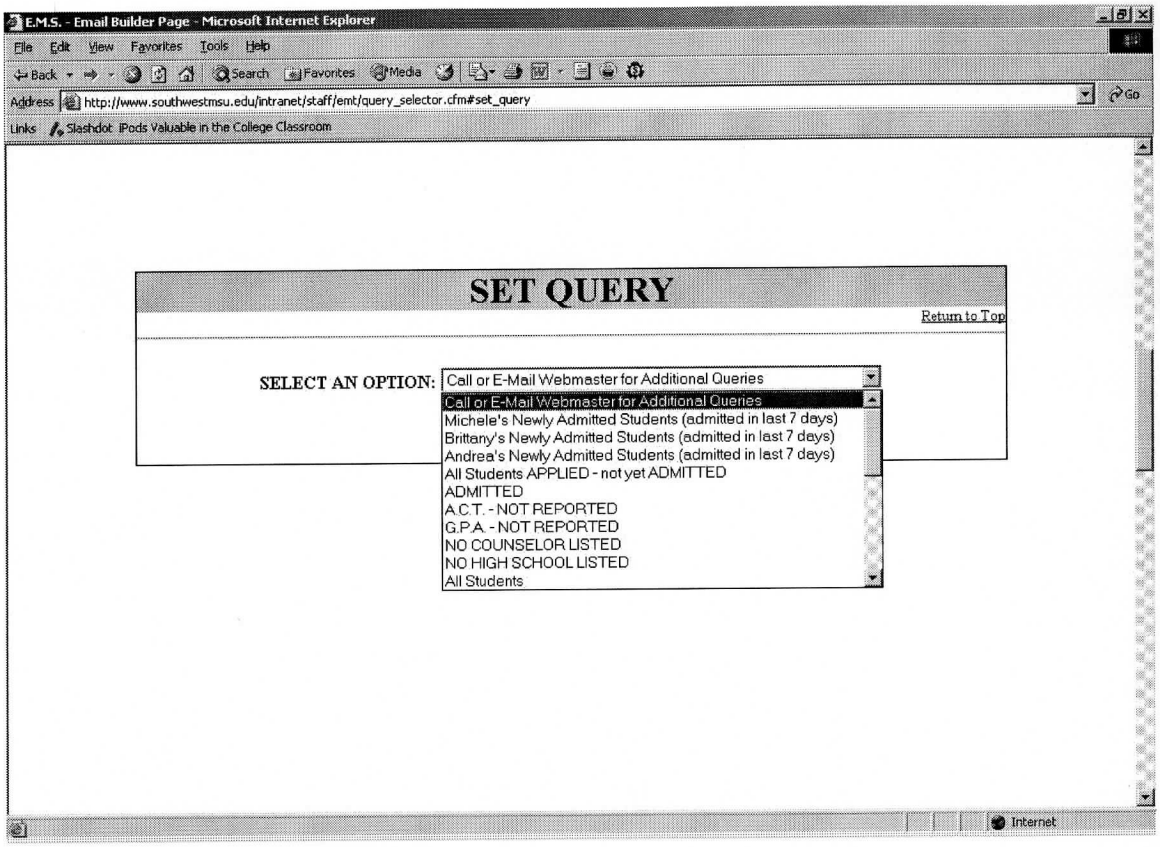


Figure 3.4 Set Query Dropdown

Coldfusion Set Query Dropdown Code Sample:

```
<select name="query_value">
        <option value=" where (counselor_name = 'Michele' or HSInstId in (select hsinstid from
        emt_territory where tech_id = '00111392')) and (admdate between '#olddate2#' and
        '#current#')">Michele's Newly Admitted Students (admitted in last 7 days)
        </option>
        <option value=" where applieddate <> " and admdate is null and (stop_cntct_code is null or
        stop_cntct_code <> ")"> All Students APPLIED - not yet ADMITTED
        <option value="where admdate <> "">ADMITTED
        <option value="where actscore = 0.0 ">A.C.T. - NOT REPORTED
        <option value="where hsgpa = 0.0 ">G.P.A. - NOT REPORTED
```

By defining the SQL statement in the option value of the dropdown select box, it is possible to send a query dynamically through a selection process. The select box is part of a form that submits to a page, which uses the form value in a dynamic query that routes a user to the display, which allows them to email that group of students. The ability to define complex queries in the set query section allows flexibility of how the user data mines students. If the query is not as complex and does not require as many data elements, the user can then use the "Build Your Query" function.

Build Your Query

The ability for the user to take control of their database lists is vitally important to success (Shearer, 2004). Independence from the technical support staff both reduces support time and aids in the independence of an admissions counselor to manage their own territory of students.

Step 1 Select Student Group

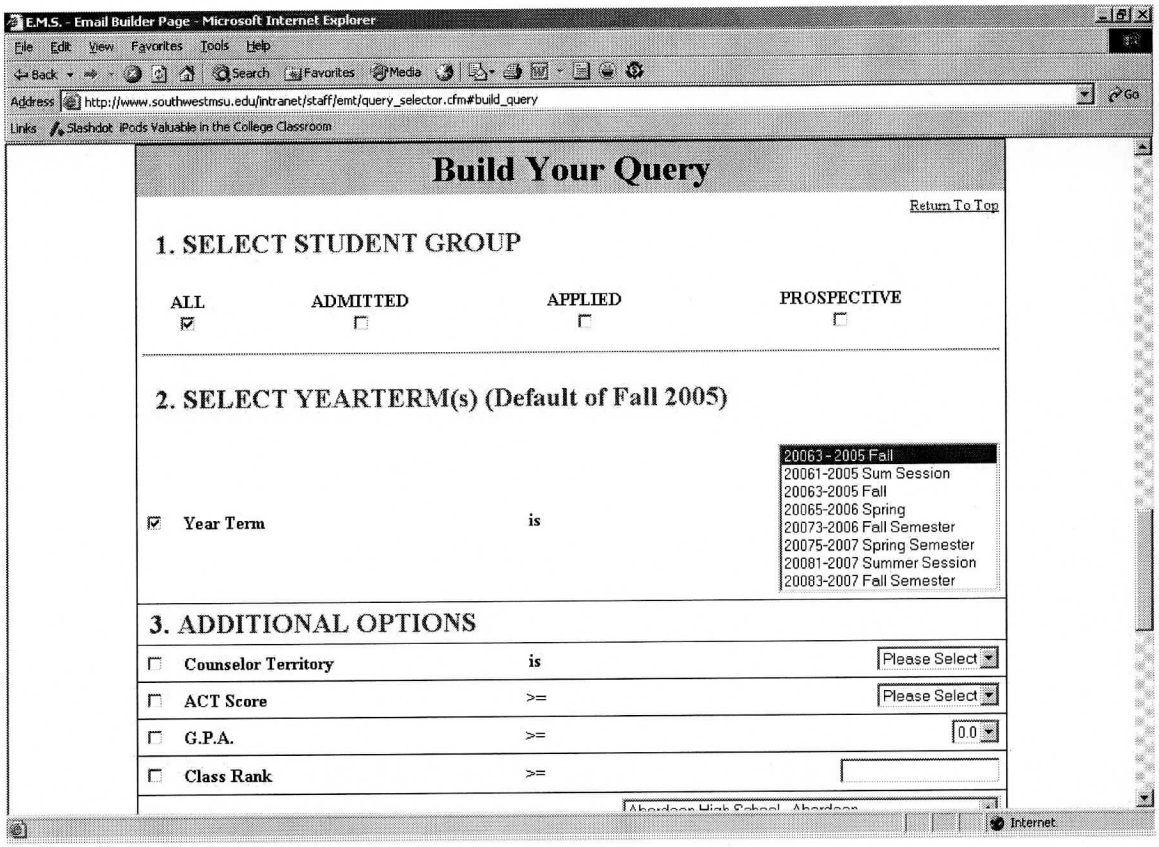


Figure 3.5 Build Your Query Page

1. SELECT STUDENT GROUP
    a. The first duty of a user when using the "Build Your Query" is to decide what basic student criteria will be used to define the batch of students.

The students are categorized in four types: Admitted, Applied, Prospective and All. The "All" category is completely independent of all factors that label students into the "Applied", "Admitted" or "Enrolled" categories. The prospective pool is a group of students that have not yet applied or been admitted. It requires the fields to be null or blank. The "Admitted" checkbox requires that the admitted field have a value regardless of whether the applied field is filled out. The "Applied" field means that the applied type has to have a value at the same time the admitted field is blank. This first section of code groups the students in a large batch.

2. Select Year Term

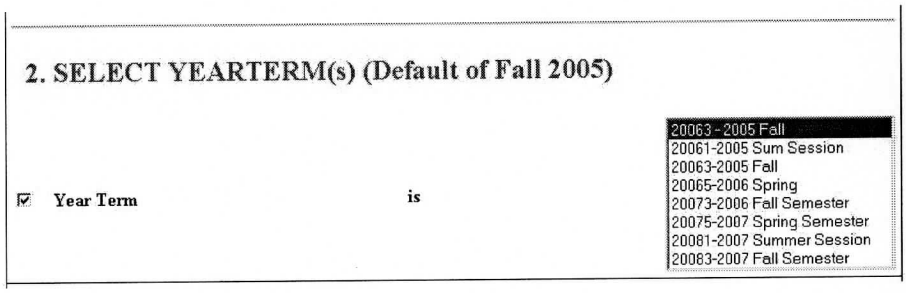


Figure 3.6 Select Year Term

a. The second choice for the "Build Your Query" is to select the year term to narrow the search. This is a multiple select box that is built from a query of the prospective student data table so that only year terms exist in the database show up.

3. Additional Options

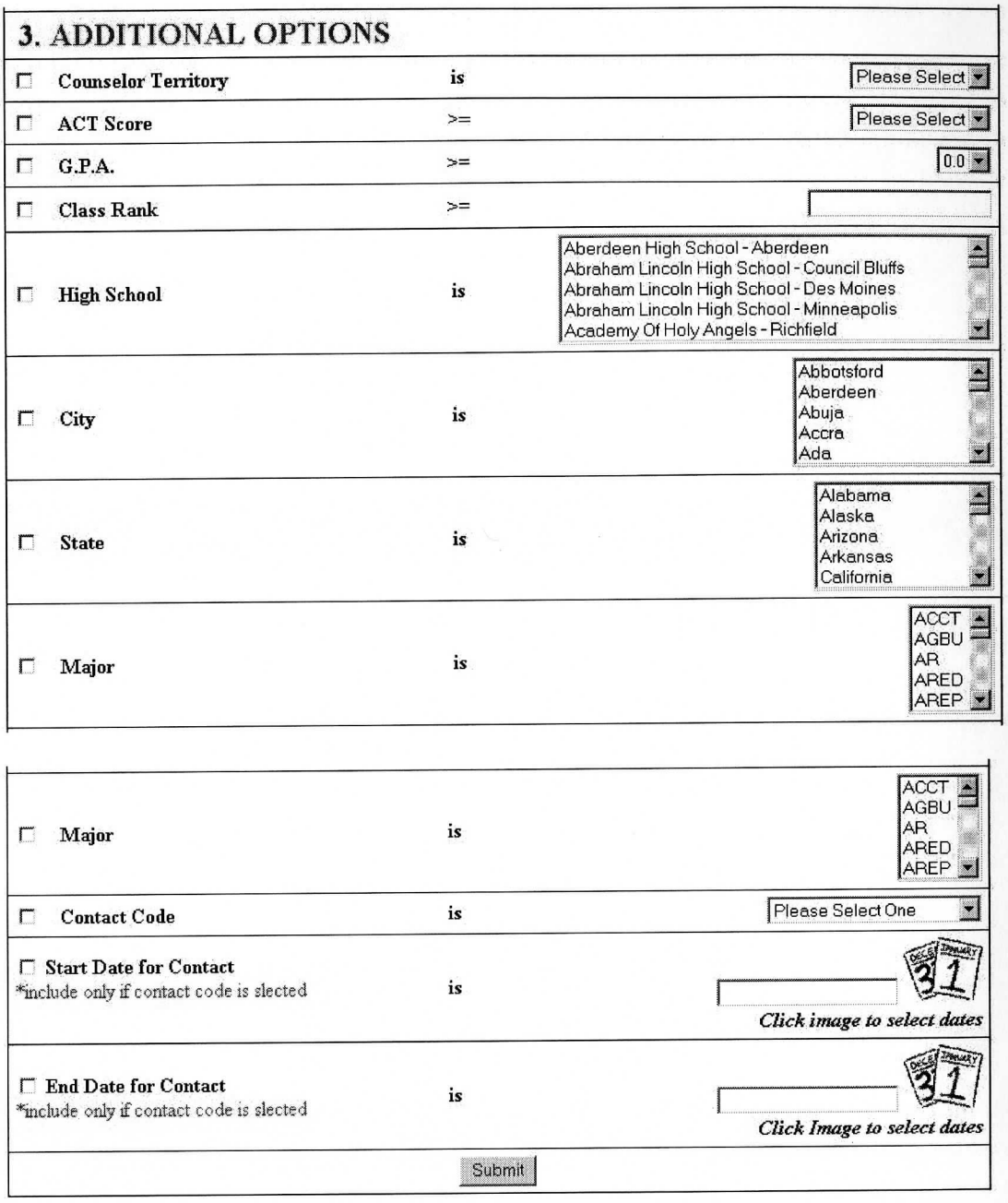


Figure 3.7 Additional Options

a. The user has the ability to select from predefined criteria by checking which attribute to include and setting the criteria in the right option boxes.

Each criterion is based off of the data downloaded through the central office. The user is unable to select any range, such as city, that does not exist within the database. This check and balance allows for error free querying.

Table 3.2: Criteria Options

Criteria	Description
Counselor Territory	Counselor Territory limits the query to those high schools that have been entered as specific to a particular counselor
ACT Score	ACT greater than or equal to a selected ACT score
G.P.A.	GPA greater than or equal to a selected GPA score
Class Rank	Class Rank greater than or equal to a selected class ranking
High School	Multiple selection box of every high school in the country
City	Multiple selection box of every city included in the prospective student table
State	Multiple selection box of every state in the country
Major	Multiple selection box of available majors
Contact Code	Dropdown box of all contacts listed at the central office data warehouse
Start Date for Contact	Beginning value of a date range of the particular contact
End Date for Contact	Ending value of a date range of the particular contact

3.7 Query Processing

Once all selections of criteria have been made and the query choices have been selected, the user submits the form to the system. The form variables are then passed to the final query selector page. This page allows the user to make the query an "And" or "Or" type query. If there is more than two variables, it will default to an "And" SQL WHERE statement. However, if only two variables are passed to the final query selector page then it is possible to switch the type of SQL statement to include an "And/Or" statement. For example, the user can state that a student must have a GPA of 3.0 and an ACT score of 20. Meaning that both of those statements must be true in order for the query to display results. If the user were to say that a student must have a GPA of 3.0 or an ACT score of 20, if either of these statements are true then results will be returned. Often scholarships will have requirements that need to have either one attribute or another to qualify. SQL queries that include "or" statements fit these types of inquiries better than other types.

A list must be created from the form values selected on the previous page in order to create a dynamic query. The Query selector page first checks to see if a set query has been sent or a built query. Once this has been established, it sets the values to hidden fields in a form and establishes the "And/Or" status of the page. Once this has been accomplished it then passes the values on to a results page.

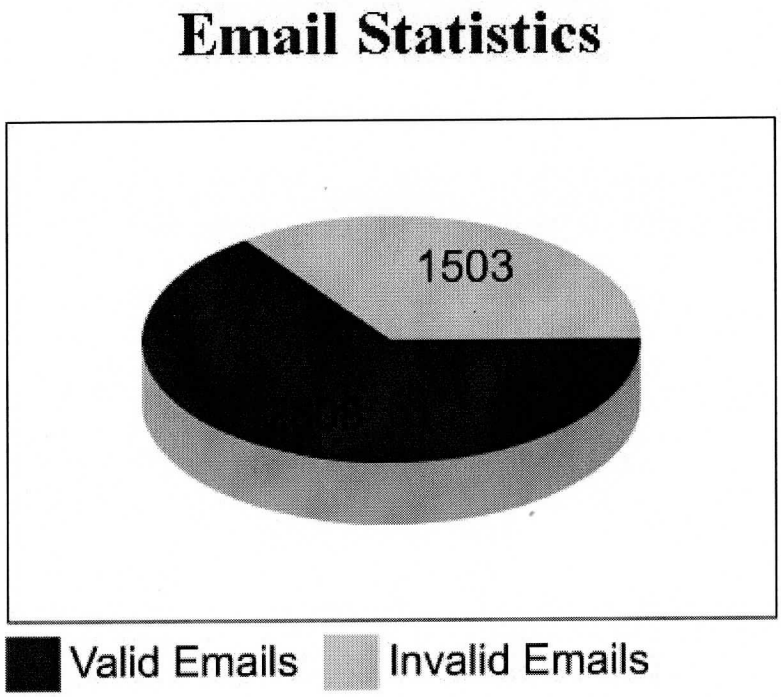
ColdFusion Query Example

```
<cfif ParameterExists(form.gpa) is "yes">
        GPA  greater than #form.gpa_value#
        <input type="hidden" name="gpa_value" value="#gpa_value#">
        <input type="hidden" name="gpa" value="#gpa#">
        <cfif c gt 1 and r is 1 or h is 1 or ci is 1 or s is 1 or m is 1 or cc is 1>
         <select name="option3">
          <option>And</option>
          <option>Or</option>
         </select>
        </cfif>
```

3.8 Results, Export, and Email

Results

Once processed through the forms, the data is outputted to the user for evaluation. This screen allows the user to evaluate the group that they have queried out of the system. The user is greeted with a graph depicting the number of students that have email addresses listed in the system and those that do not have email listed.



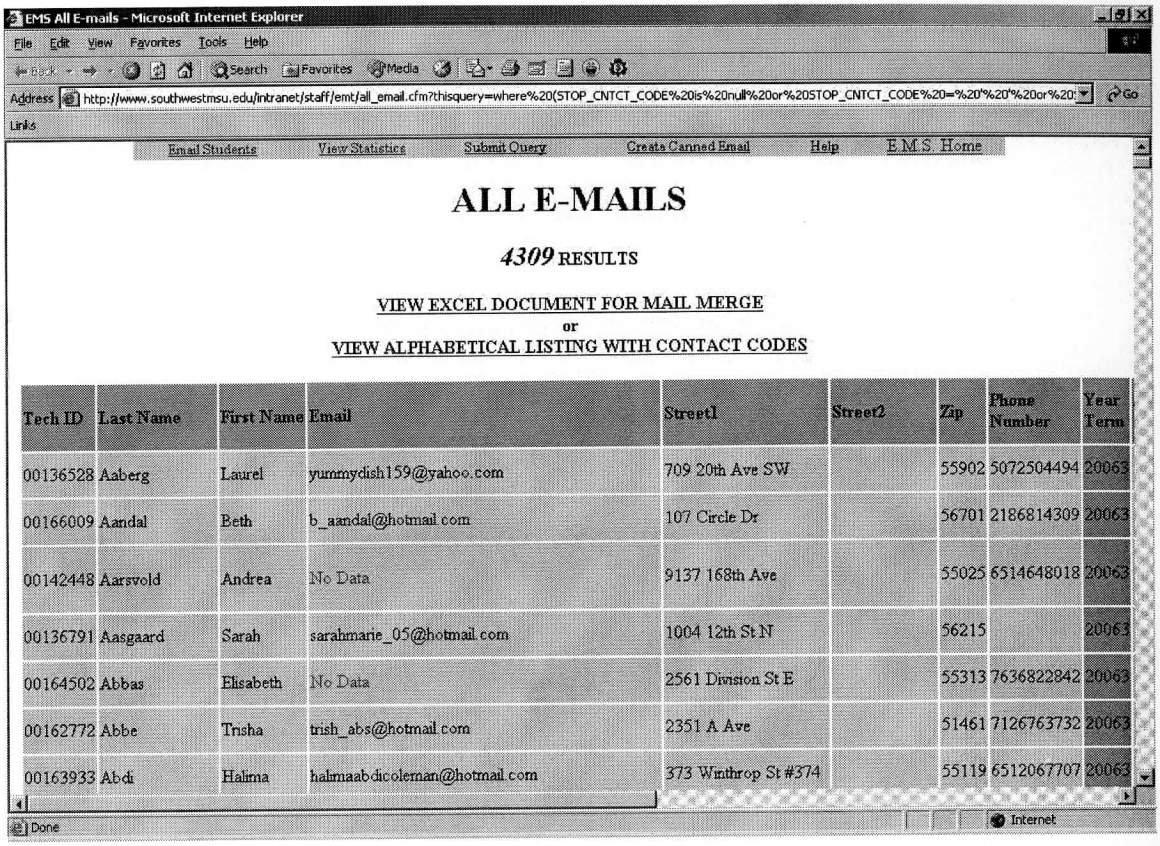
Results	Link
4311	Total Matches
1503	INVALID E-Mail Addresses
2808	VALID E-Mail Addresses

CLICK HERE TO CREATE E-MAIL

Figure 3.8: Email Statistics

Export

From this point a user has two options for continuing; they can either validate the user list by clicking on the links to view total matches, invalid email addresses, valid email addresses or they can create an email. When clicking on any of the links you are met with an output of the results for that particular query. Basic information is included such as names, address, telephone, links to contact information and other statistics. Each option has the link provided to export the list to a document format and an excel document for a mail merge. The ability to mail merge is key to delivering a solution to communicating with those students that have not provided their email address to the admissions office. A key design issue was building a system that can handle all of the students that an admissions office would deal with in a central location. Providing this central location allows staff to keep track of exactly what students they are communicating with.



Tech ID	Last Name	First Name	Email	Street1	Street2	Zip	Phone Number	Year Term
00136528	Aaberg	Laurel	yummydish159@yahoo.com	709 20th Ave SW		55902	5072504494	20063
00166009	Aandal	Beth	b_aandal@hotmail.com	107 Circle Dr		56701	2186814309	20063
00142448	Aarsvold	Andrea	No Data	9137 168th Ave		55025	6514648018	20063
00136791	Aasgaard	Sarah	sarahmarie_05@hotmail.com	1004 12th St N		56215		20063
00164502	Abbas	Elisabeth	No Data	2561 Division St E		55313	7636822842	20063
00162772	Abbe	Trisha	trish_abs@hotmail.com	2351 A Ave		51461	7126763732	20063
00163933	Abdi	Halima	halimaabdicoleman@hotmail.com	373 Winthrop St #374		55119	6512067707	20063

Figure 3.9 Information Report Screen

Email

One of the last options available to the user is the ability to email the group of students. The core operation of the enrollment management system is to communicate to students electronically and the ability to create html email is key to success. The design of the enrollment management email section allows a user to choose from a list of available options in order to spice up the communications.

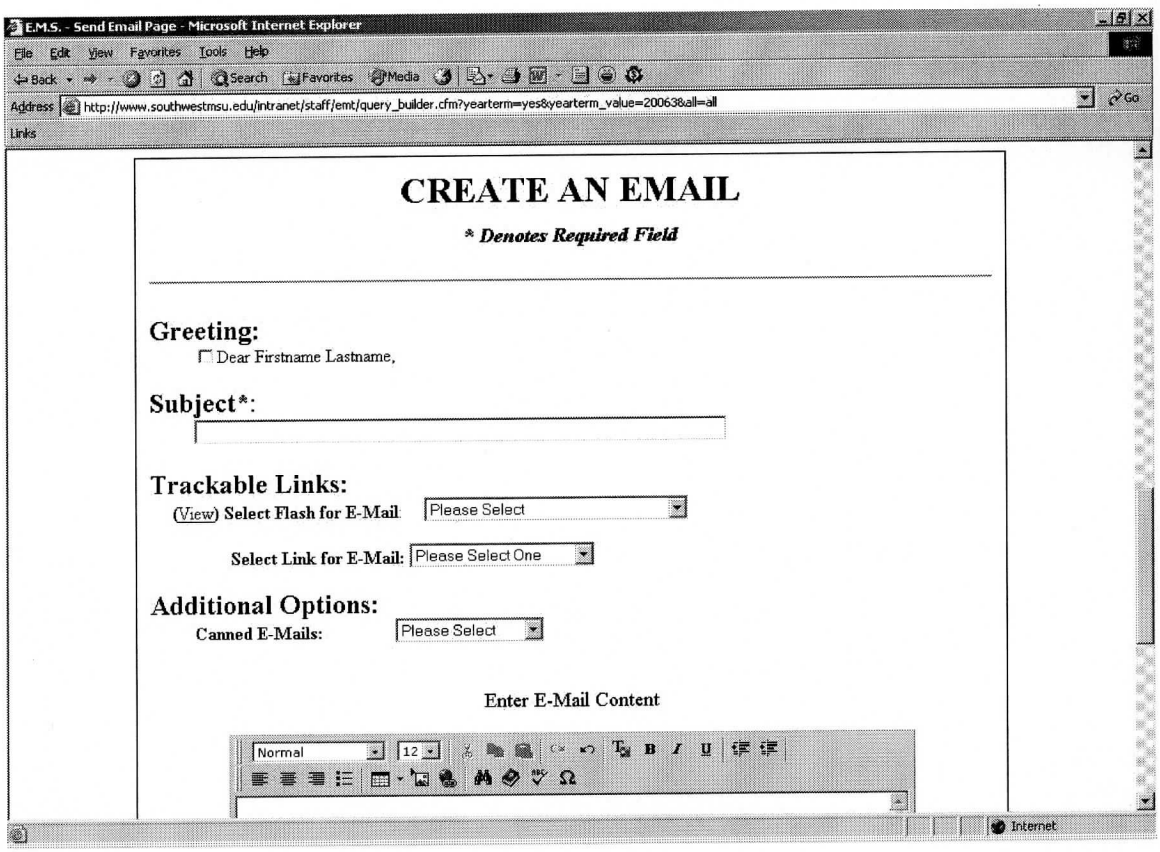


Figure 3.10: Create Email Screen

Table 3.3: Email Additional Options

Greeting	Personalize email to the name of the student in which the email is going to
Subject	Required field for subject line of email
Tracked Links	1. Flash email link inserted into email that is a link that becomes tracked to those that click on it. 2. Link to admissions office website that is tracked.
Canned Emails	Set text emails that auto insert into the body of the message. Used for communication that remains the same year after year.
E-Mail Content	Active Edit window that allows full Microsoft Word features for email creation

3.9 Territory Management

Southwest Minnesota State University's enrollment office has several admission counselors on staff. The organization of their office and the wide region they serve dictates that they separate the map into territories. Once the region has been identified, the counselors begin to take note of what high schools are in that region. The enrollment management system needed to be able to make these same separations of students so that the counselors could have the ability to control their queries. The solution to this management was developing the Territory Management module. In this portion of the software the user is able to add their territory manually. The first step to accomplishing this is to click the "Add Territory Page" link. This page lists all of the states and nations that exist in the MnScu data system. Once the counselor has selected those states they are presented with a list of high schools in that region. They can select multiple high

schools and add them to the territory. This ability to add and subtract to the territory, populates the build your query page and is an element that allows limits to the queries.

Territory List Page

The territory list page will show a user their particular query with the ability to add and remove each high school.

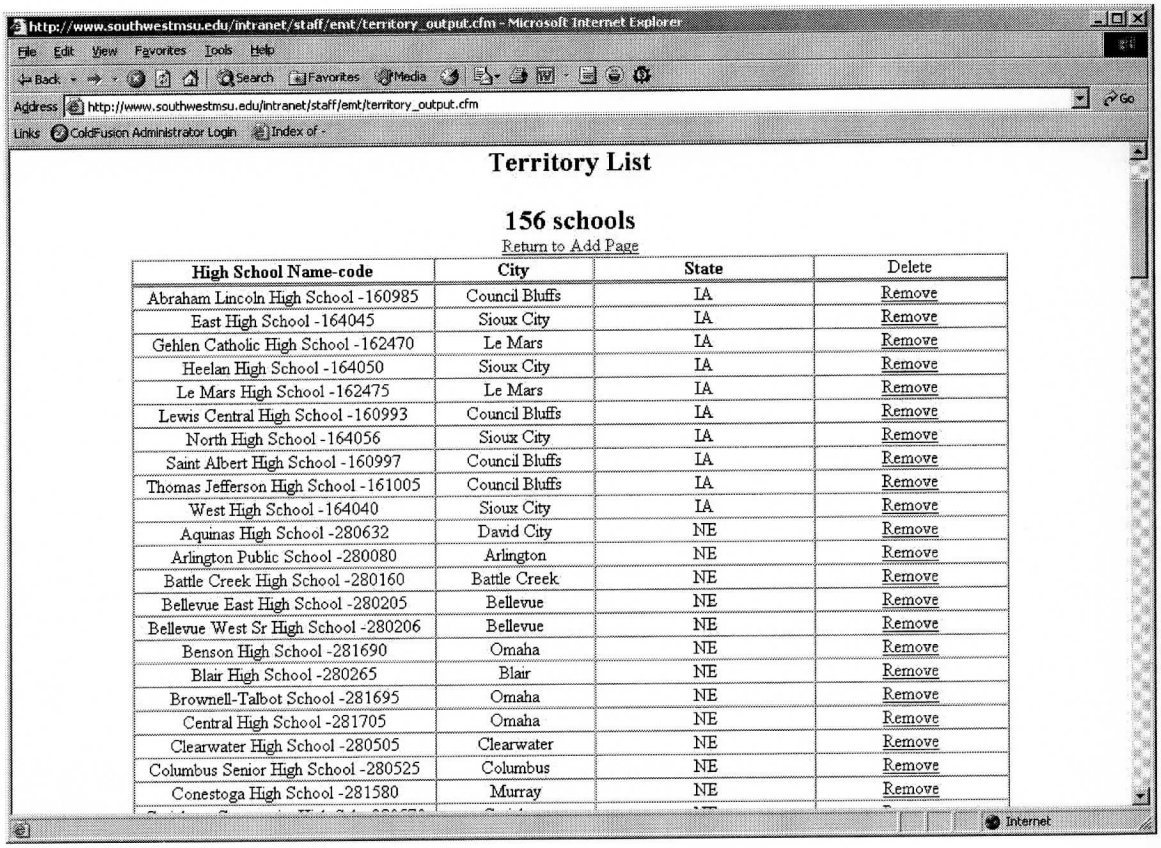


Figure 3.11: Territory List

Select a State

The territory management screen will direct a user to add a high school. When the user clicks on the add page they are greeted with the ability to select the state the high school resides.

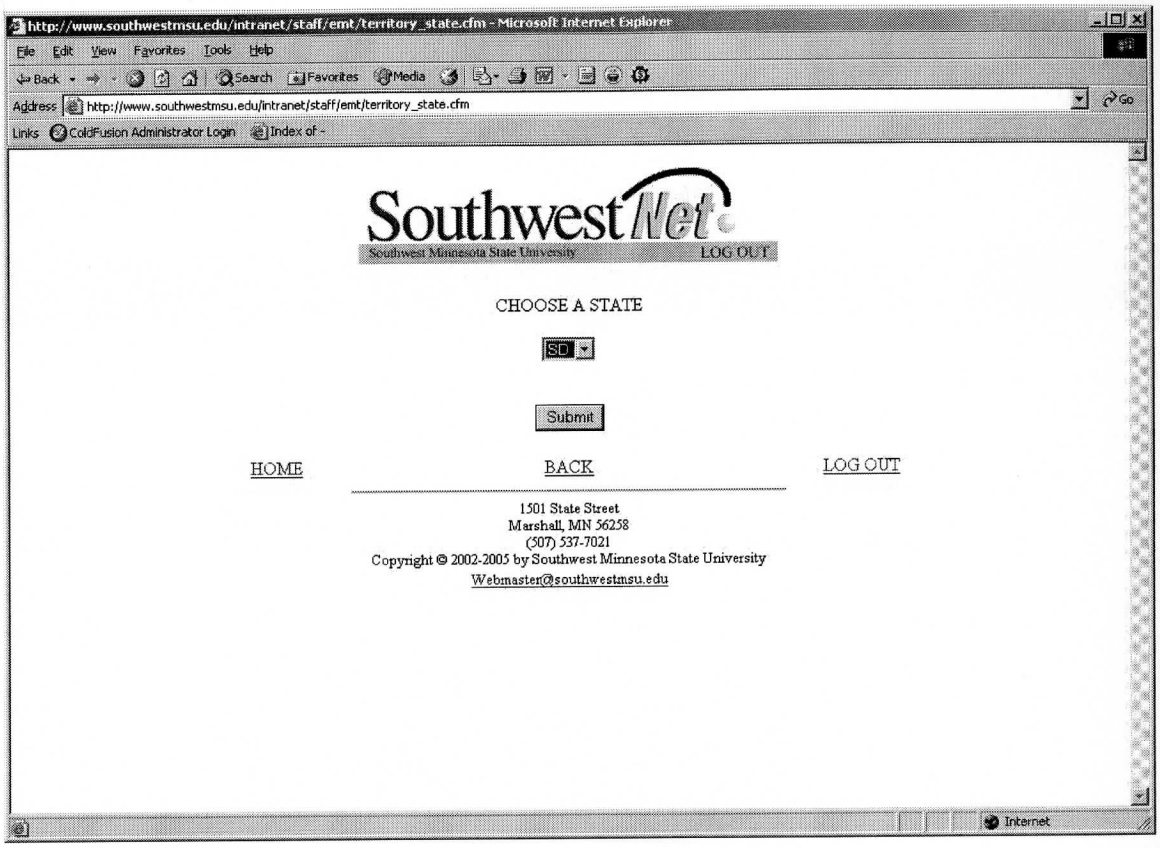


Figure 3.12: Select a State

Choose your High School

The user will be directed to this page after they have narrowed down which state the high school exists. This page queries the database to display all high schools in the database from the particular state, which has been passed to this page as a form variable.

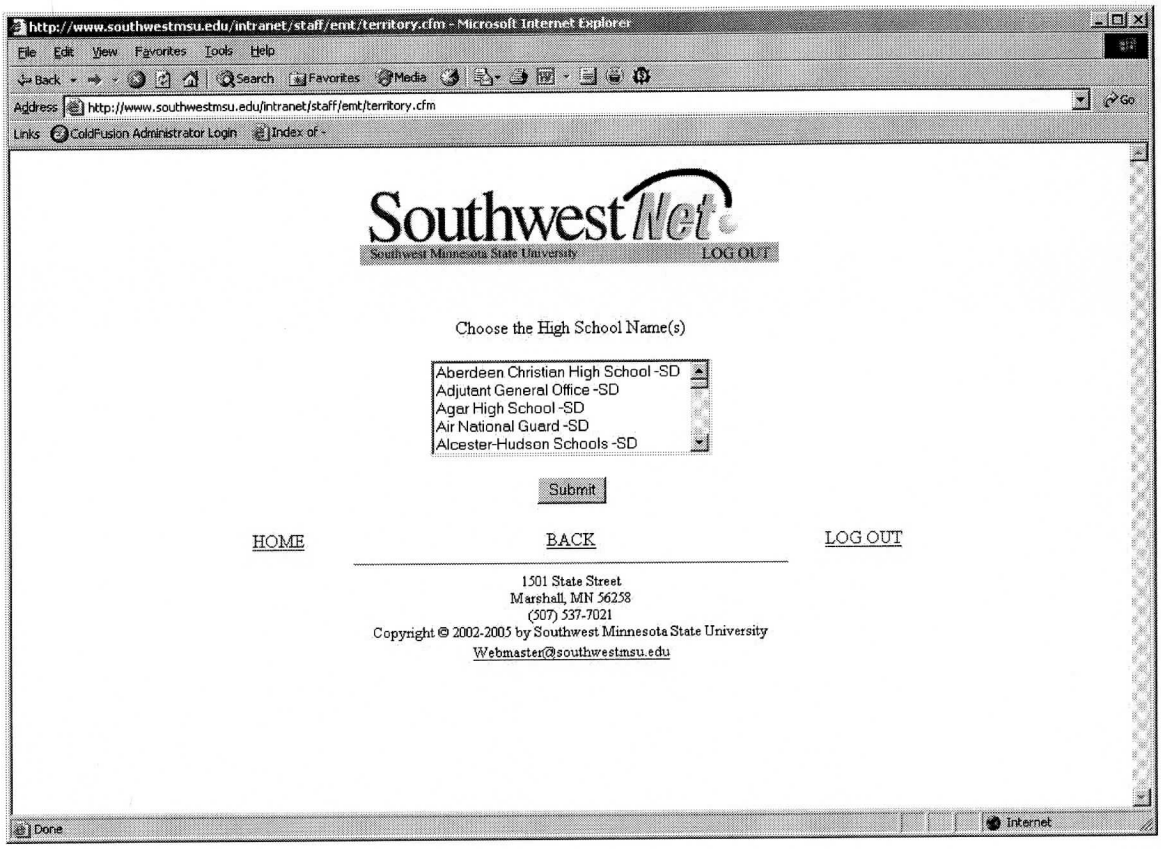


Figure 3.13: Select a High School

3.10 Counselor Reports

Control and success of the system was a key element through planning. Current issues in the enrollment office included the lack of statistics and the ability to react to changes in information. A successful system would need to pass control of information to the counselor level. Addressing this problem, the counselor reports were created.

These reports allow the counselor to get a snap shot of key information that can help them be effective in managing their territories. The first option is to "View Students With Contact Info". This dropdown box allows them to obtain a list of every student in their territory along with each contact from the EMT_Contacts table. This provides a history of contacts to a particular student. Counselors use this when a student is on campus for a visit or if returning a phone call and would benefit from seeing how the admissions office had handled that student. Common information includes items such as number of phone calls and if they have participated in a campus visit. Another built-in function of this report is the ability to view the success rate of the emails sent through E.M.S. In the dropdown titled "View Tracking Results" the counselor views a date sorted list of previous emails by subject and name. They then can choose to view an email and see how many people clicked on the links provided through the email system. The counselor is shown a chart depicting the total number of students the email was sent to, along with the number of people that clicked on a tracked link. Other information accompanies this, such as the content of the message sent and the email id given to the database entry. The final section of the report is a pair of charts that list emails sent and students contacted.

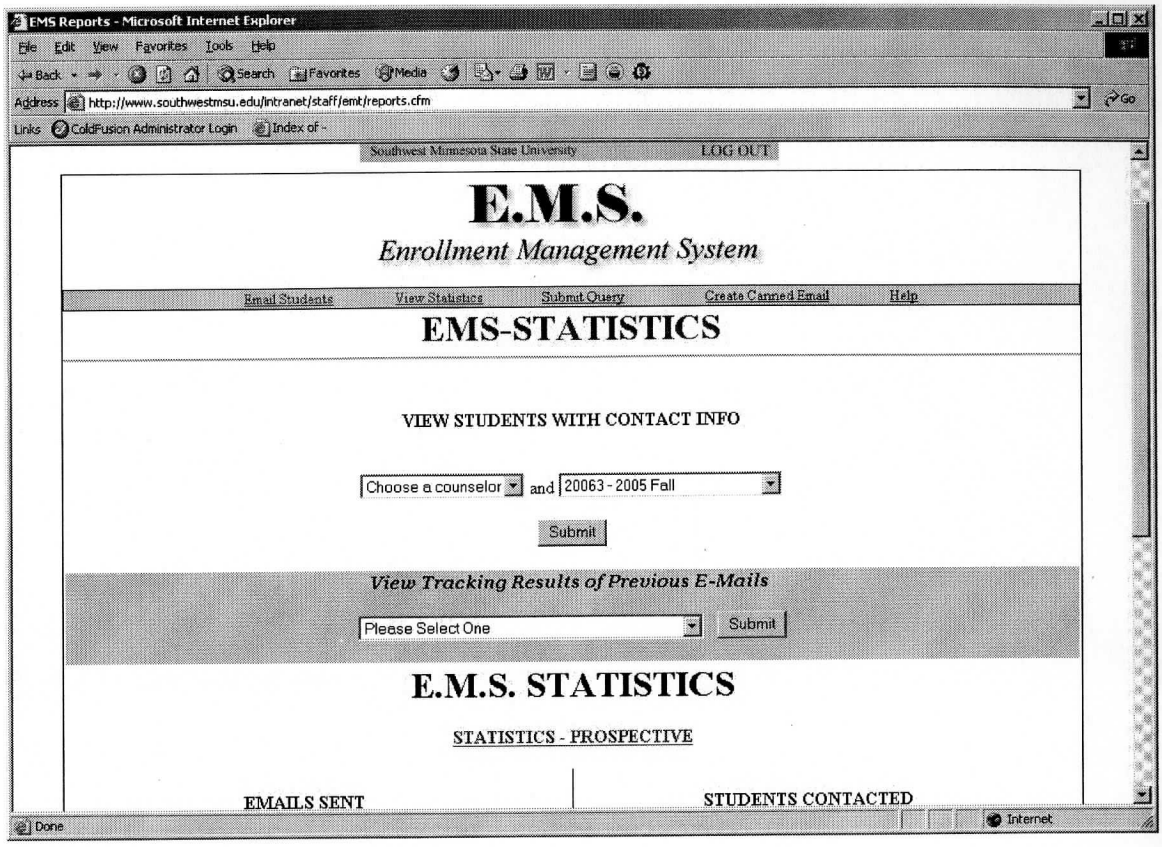


Figure 3.14: E.M.S. Statistics

The numbers of emails sent and students contacted are represented in the following charts. These charts show how often the software is being used. Total emails sent allow the administrators to see how many batch email groups have been queried. This chart also shows the user their own progress against the total number of system emails.

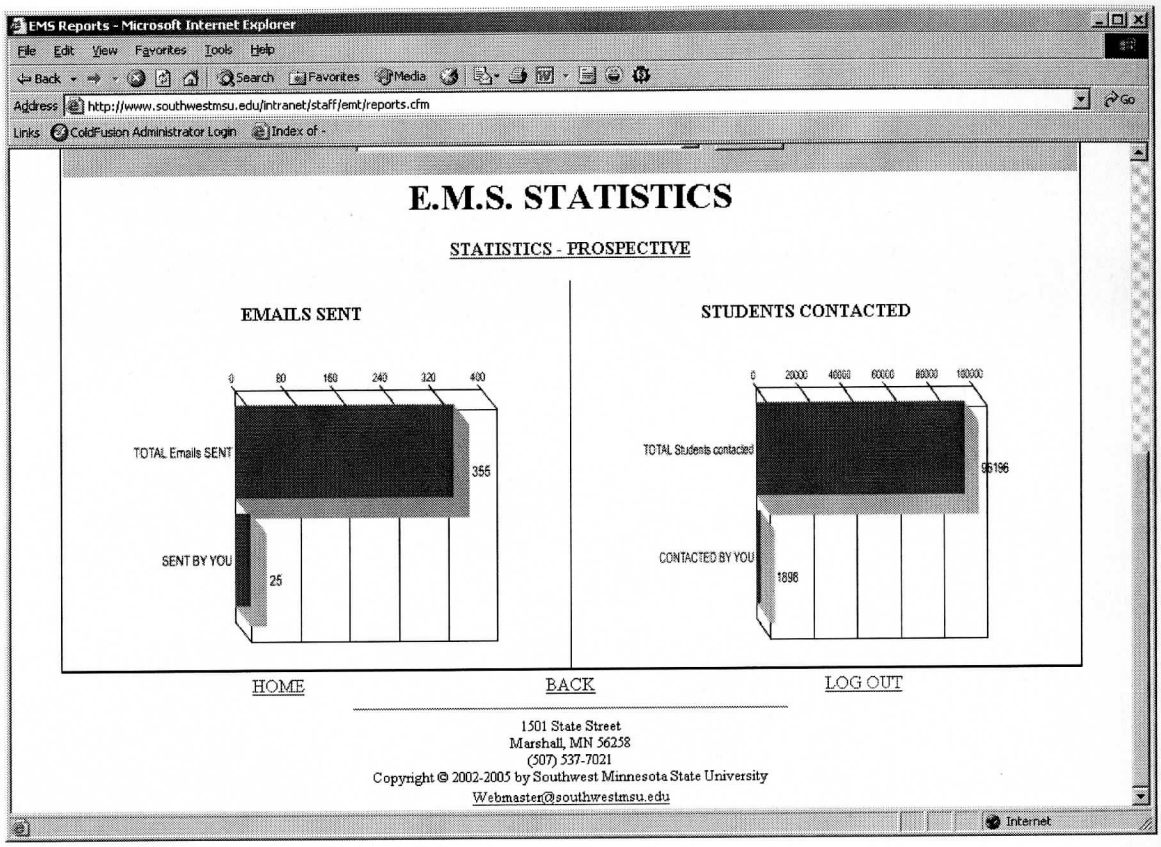


Figure 3.15: E.M.S. Statistics Charts

When a user views the contact code history page, they will see a complete listing of all contacts of each student in their territory. This page is vital to a counselor's success, since often a student may call or drop in and the counselor will have to refer to previous experiences in order to make the student feel as if they have had a personal relationship with the University.

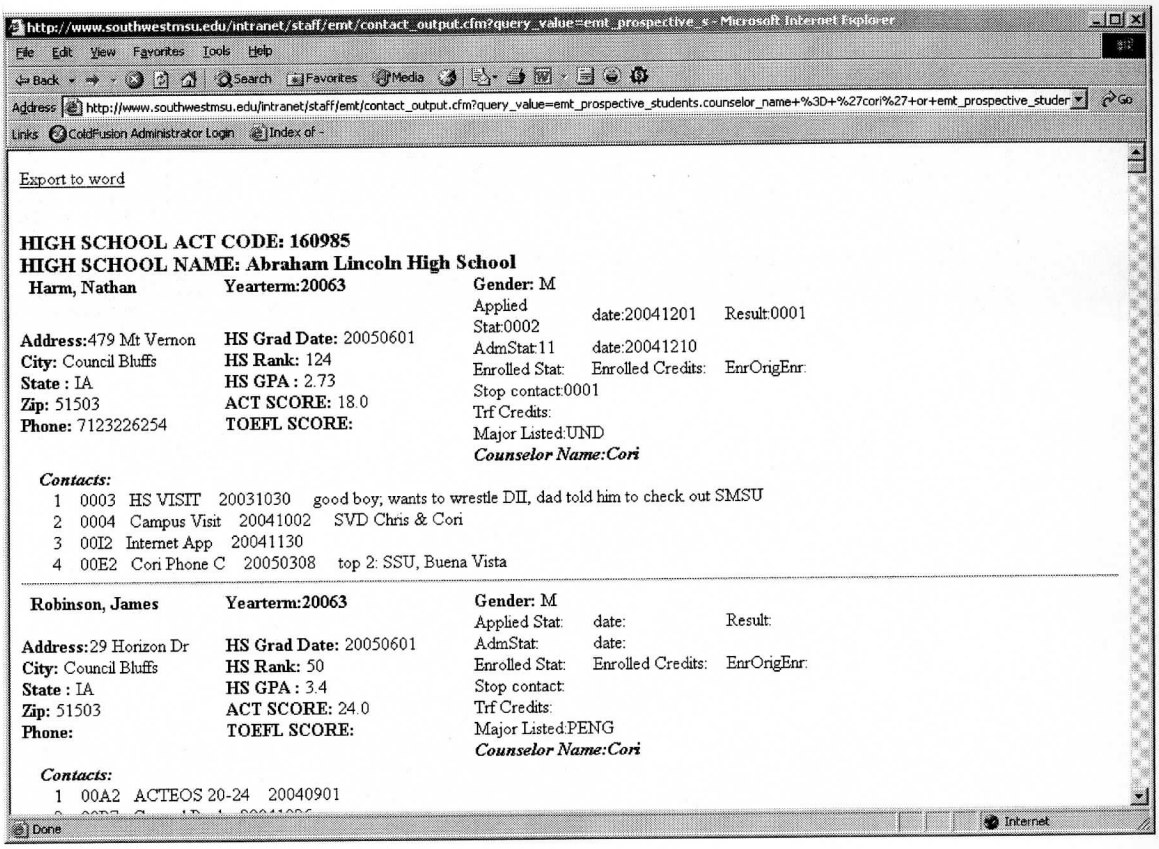


Figure 3.16: Listing of Contact Code History

The email statistics page allows the counselor to view the specific statistics of any particular email sent by them. They can also view how often the student clicked on the tracked links. These statistics help the counselor to fine-tune their approach in order to be more effective.

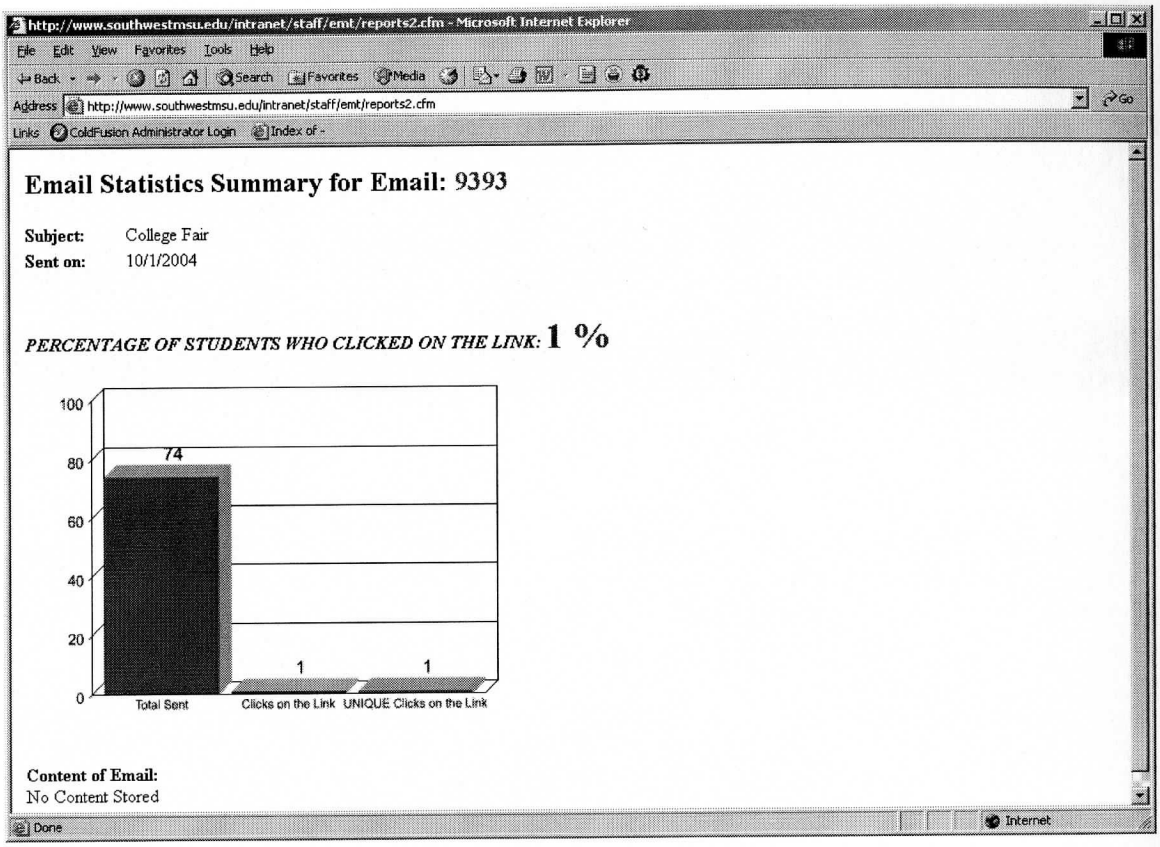


Figure 3.17: Email Statistics Page

3.11 Enrollment Director Reports

The director of enrollment services oversees many counselors that contact students through E.M.S. The director is provided a snapshot of the progress each counselor. This allows the enrollment director to compare effectiveness of enrolled rate versus production level in the system.

The enrollment director is allowed to view counselor stats by student or high school. These charts aid the director with distinguishing between those counselors who have a significantly higher workload than others.

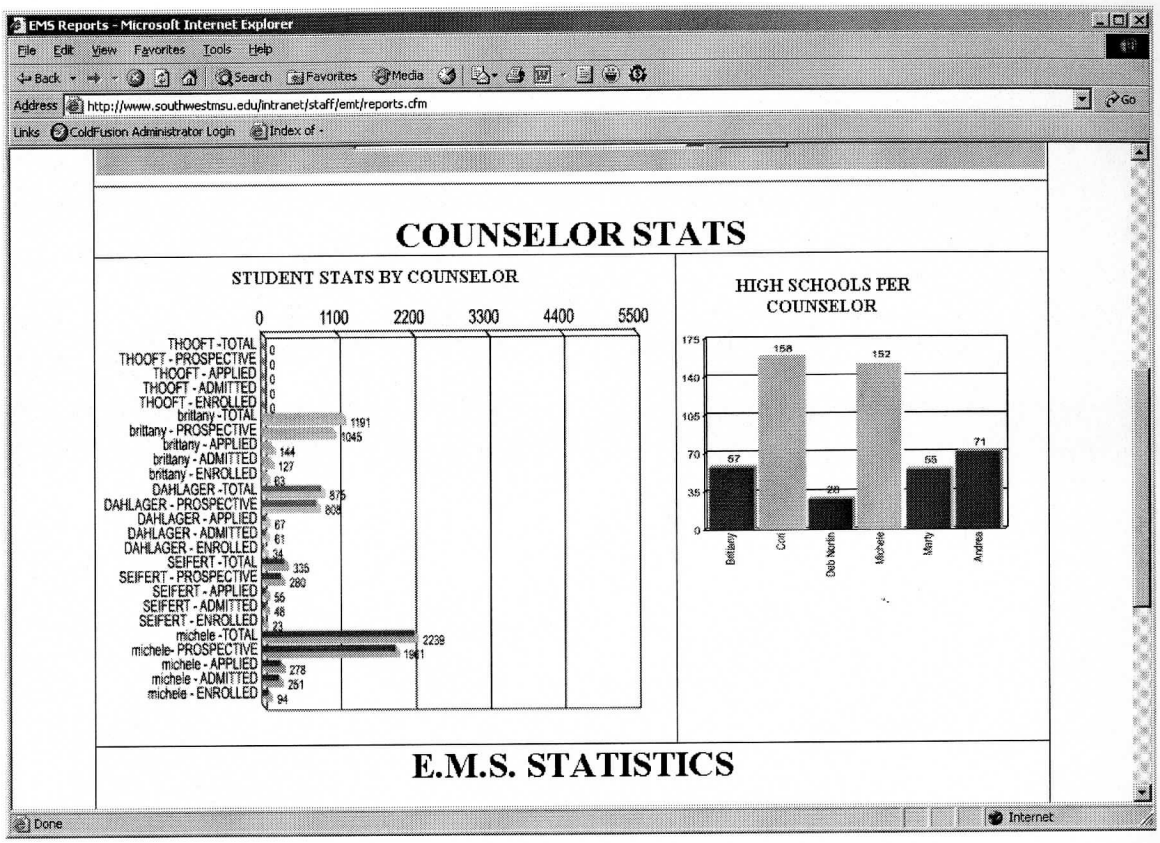


Figure 3.18: Counselor Stats for Enrollment Director

The charts shown below demonstrate the total emails sent by each counselor in the admissions office and also the total number of students contacted. These charts show the activity level of each counselor.

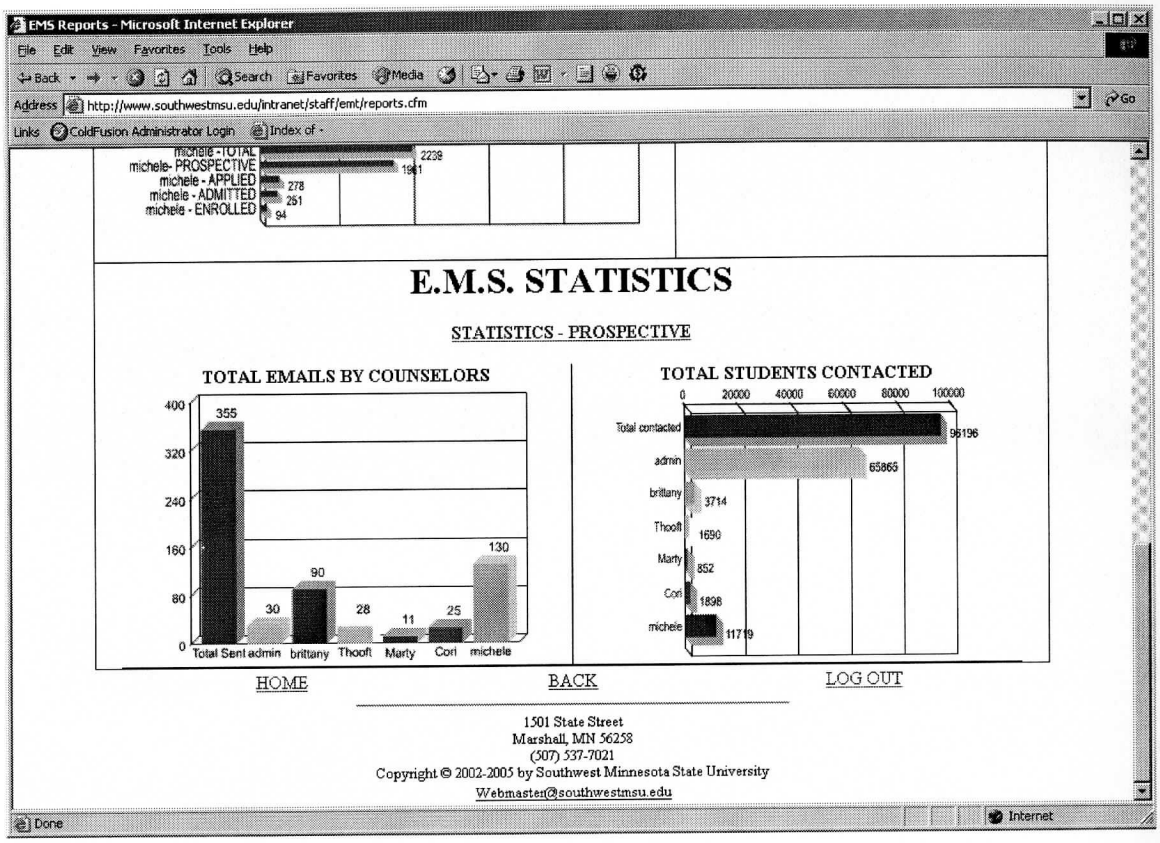


Figure 3.19: Statistics for Enrollment Director

General statistical analysis of the breakdown of prospective students is provided in chart form to the Admission office. The office can use this data to keep track of what the prospective pool is comprised of. The charts include Student Categories, Student By Major, Students By State, Ethnicity, Gender, Class Rank, and GPA.

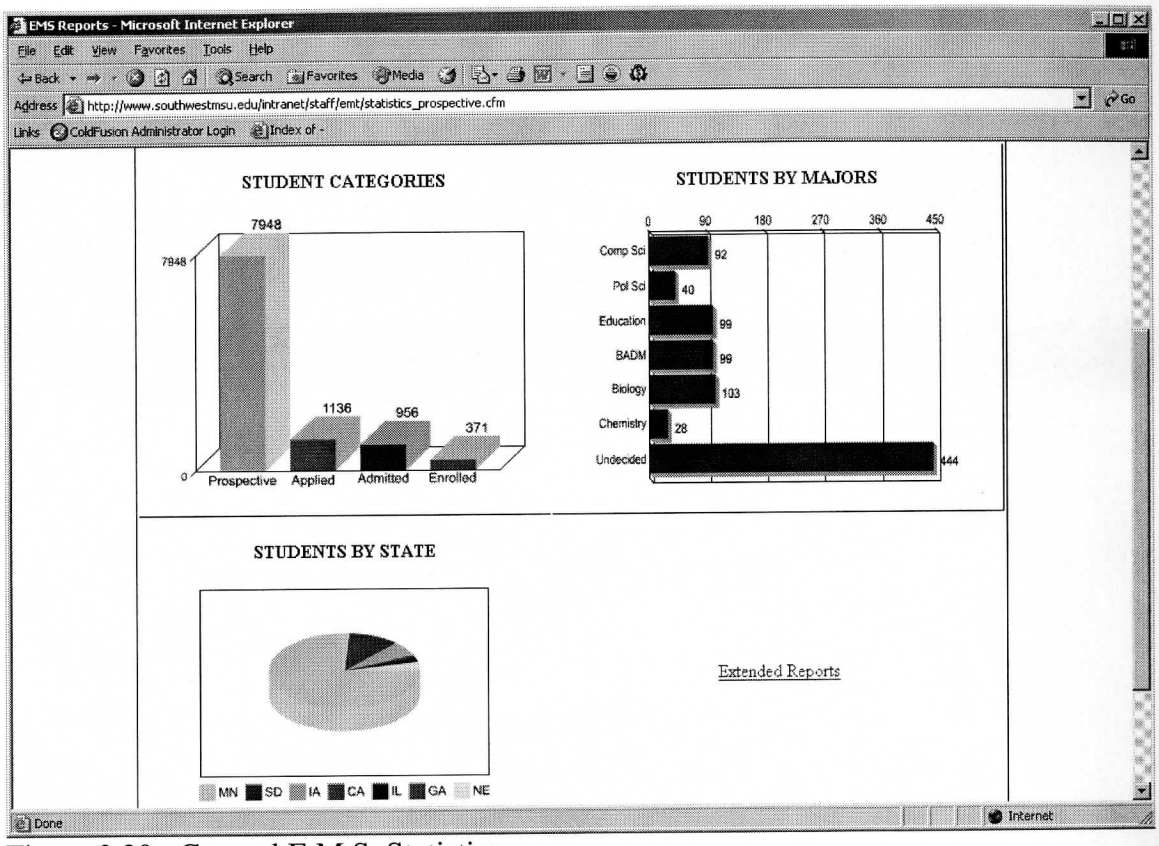


Figure 3.20: General E.M.S. Statistics

The ethnicity and gender representations allow the admission office to develop a picture of how the prospective student pool is comprised. This allows the office to hit targeted goals regarding these statistics.

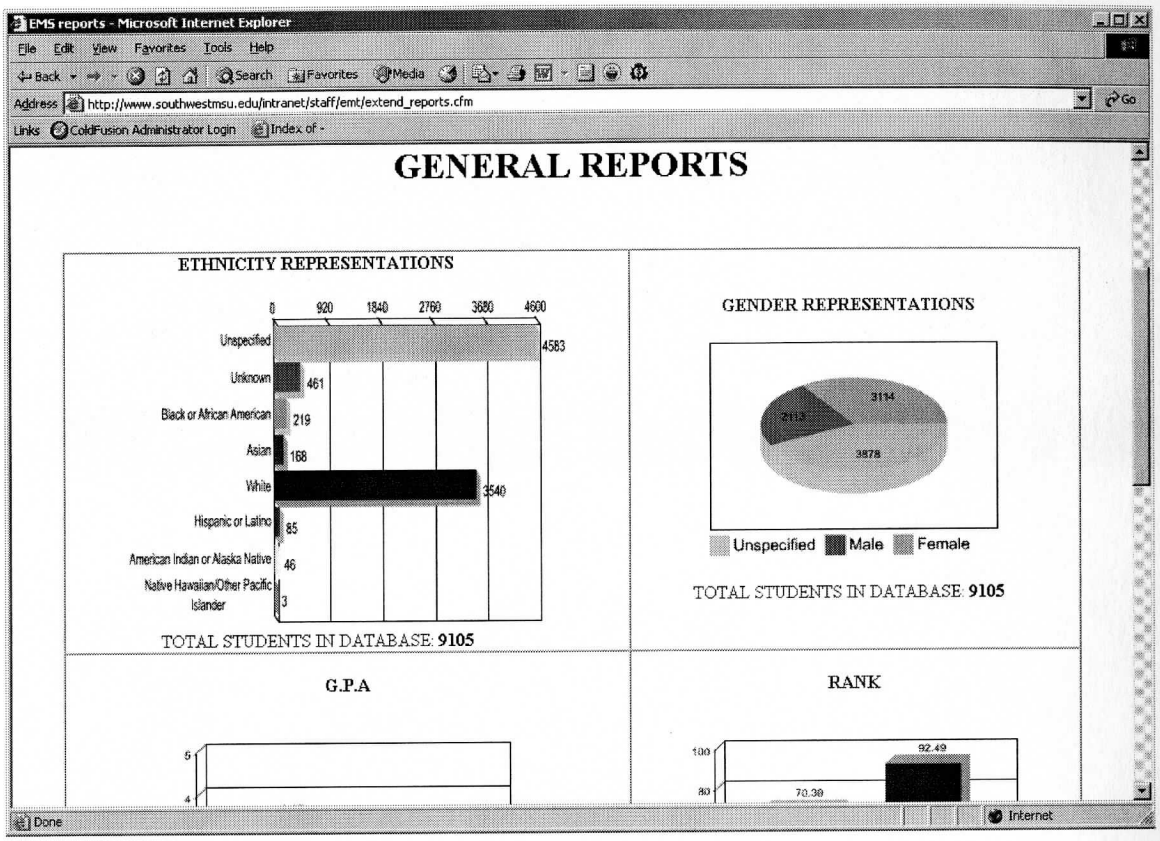


Figure 3.21: General E.M.S. Statistical breakdowns

GPA, Class Rank, and ACT scores allow the admission office to gauge the quality of the students they are persuing. By comparing the students in the prospective pool with those students that have been admitted they can immediately see the quality of the future enrolled student.

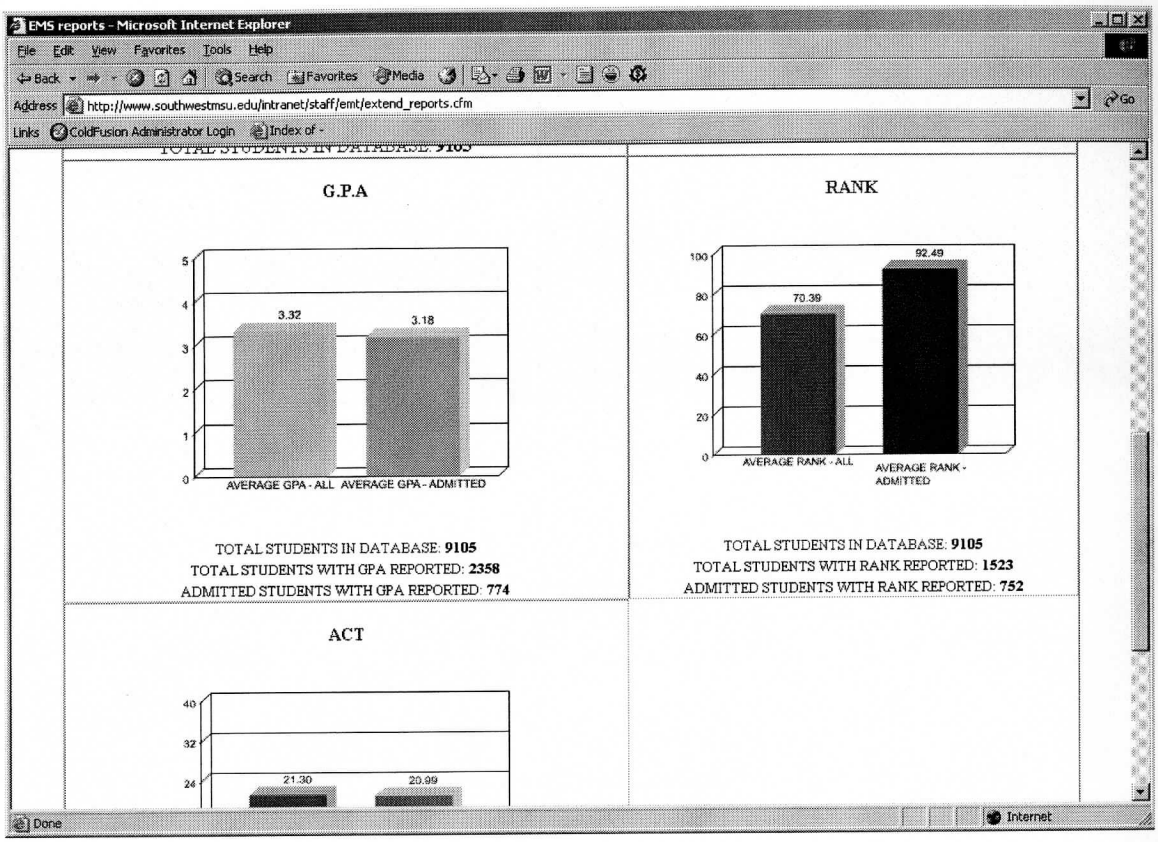


Figure 3.22: General E.M.S. Statistical Breakdown Continued

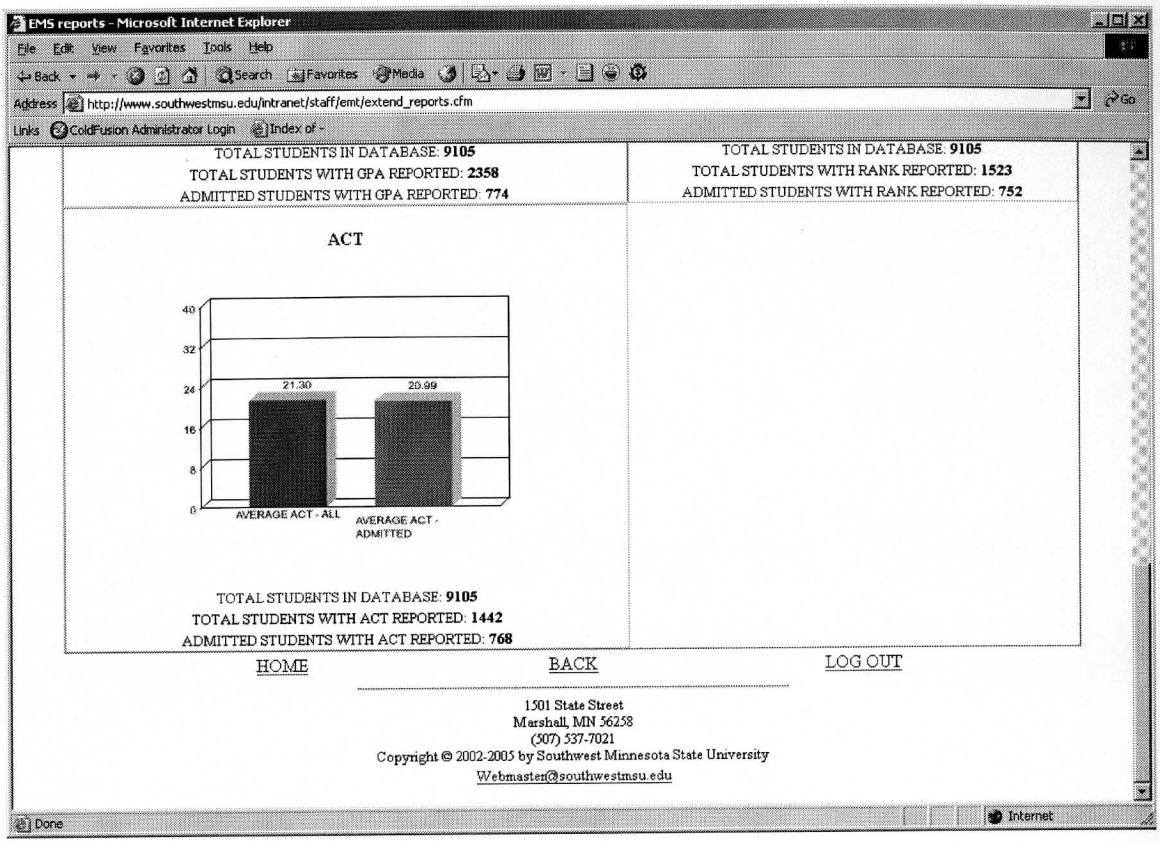


Figure 3.23: General E.M.S. Statistical Breakdown 3

3.12 Stop Contact List

One concern for developing an electronic communications tool to contact prospective students is the risk in loosing students who do not respond to this communication method (Steele, 2002). It became clear that there must be a system available for those students who did not want to be emailed, to opt out. Each email sent to every student is setup with a link to a stop contact list. This allows the student to click the link and fill out a form as to why they do not want to be contacted. The data is collected and presented on the Stop Contact Report Page. This page allows the administrator to view the reasons why this communication method did not work along with removing them from the system. By recording why a student did not want further

communications, the admissions office has been able to adapt their communications methods to be more effective. The stop contact page also allows the director to remove those students who simply do not want to be contacted. By taking those students out of the database the office saves valuable time by not having to contact as many students. This system allows a more targeted approach that returns a much higher dividend (Steele, 2002).

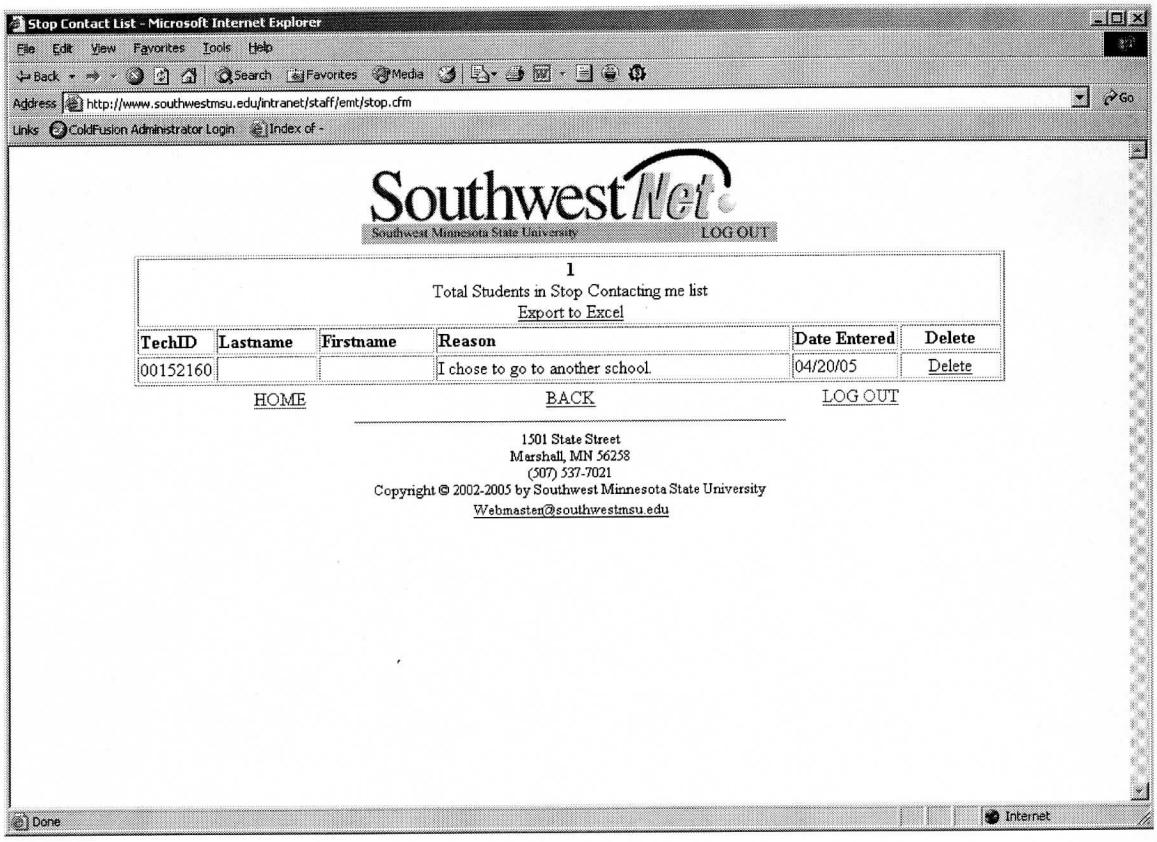


Figure 3.24: Stop Contact Report Page

3.13 Enrollment Report To Management

Accountability is a key management technique used to gauge the effectiveness of any system (Broadbent, & McDonald, & Hunter, 2003). The admissions office is accountable for their enrollment numbers to the university administration. The university administration has requested weekly updates on progress of the enrollment year (Danahar, 2004). A series of reports were created that allowed them to view enrollment data, updated daily, from their own Intranet login. The administration of the university can view current prospective, applied, admitted, and enrolled figures. They also have the ability to use the comparison charts to view the data against a historical record of where the university enrollment was on the same day in previous years. Certain areas of the tri-state region have been identified as areas that need to be monitored. The enrollment report to management allows the administration to see the breakdown of those students by area. A breakdown of available charts in the enrollment report to management follows:

The Enrollment Management Report homepage provides a quick snapshot of the enrollment statistics being generated from the Enrollment Management System. The administration utilizes this screen to monitor progress of the enrollment office.

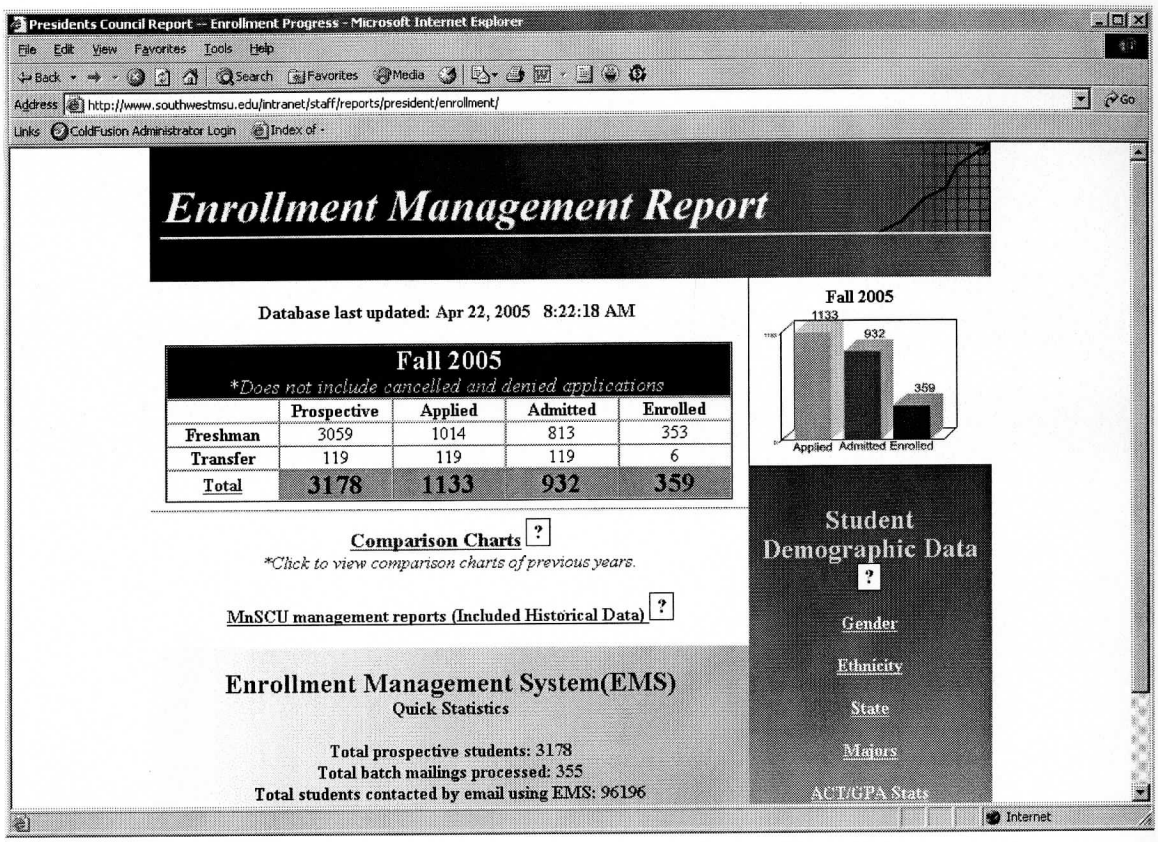
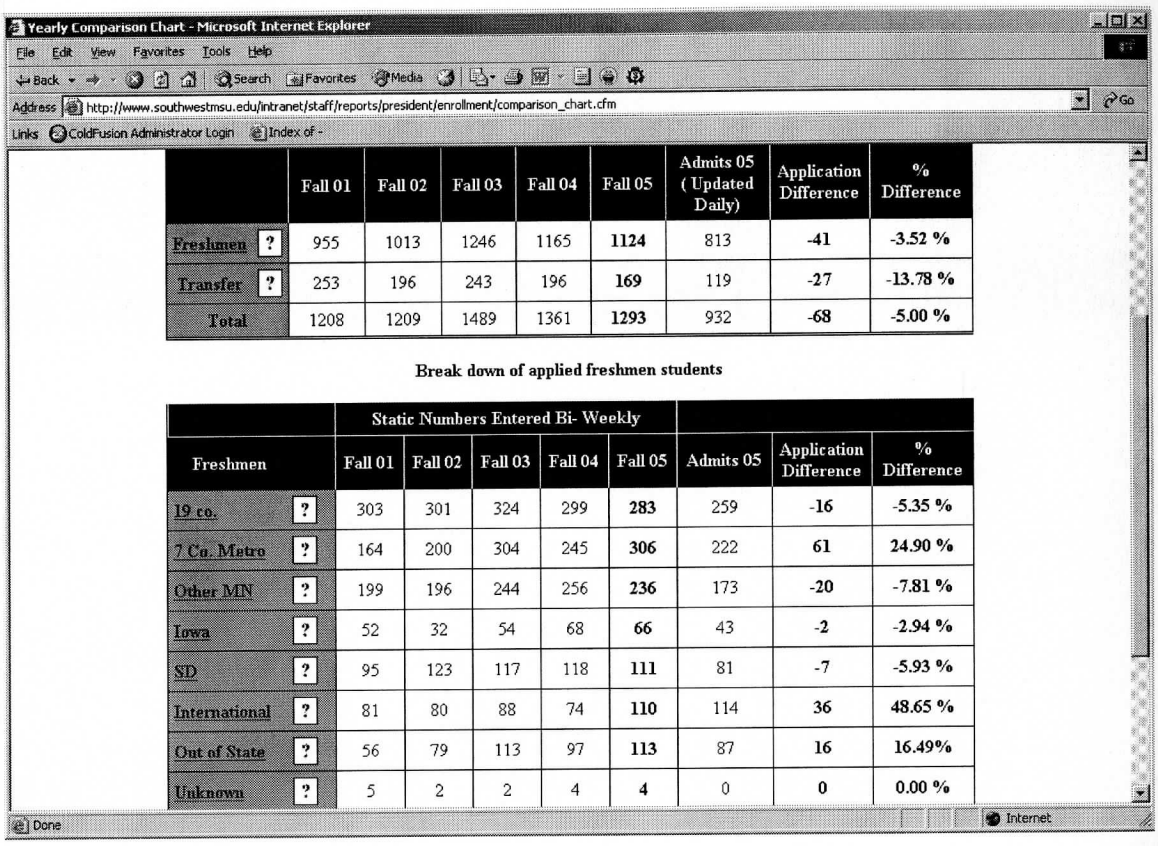


Figure 3.25: Enrollment Report to Administration

Administration keeps track of the progress of the enrollment office in specific regions of the map. They can easily identify progress made from marketing in the different student areas. Seven county metro is the counties that make up the Twin Cities schools. With marketing being expensive in these areas it is important to watch the success of the enrollment office with these groups of students.



	Fall 01	Fall 02	Fall 03	Fall 04	Fall 05	Admits 05 (Updated Daily)	Application Difference	% Difference
Freshmen ?	955	1013	1246	1165	**1124**	813	**-41**	**-3.52 %**
Transfer ?	253	196	243	196	**169**	119	**-27**	**-13.78 %**
Total	1208	1209	1489	1361	**1293**	932	**-68**	**-5.00 %**

Break down of applied freshmen students

	Static Numbers Entered Bi- Weekly							
Freshmen	Fall 01	Fall 02	Fall 03	Fall 04	Fall 05	Admits 05	Application Difference	% Difference
19 co. ?	303	301	324	299	**283**	259	**-16**	**-5.35 %**
7 Co. Metro ?	164	200	304	245	**306**	222	**61**	**24.90 %**
Other MN ?	199	196	244	256	**236**	173	**-20**	**-7.81 %**
Iowa ?	52	32	54	68	**66**	43	**-2**	**-2.94 %**
SD ?	95	123	117	118	**111**	81	**-7**	**-5.93 %**
International ?	81	80	88	74	**110**	114	**36**	**48.65 %**
Out of State ?	56	79	113	97	**113**	87	**16**	**16.49%**
Unknown ?	5	2	2	4	**4**	0	**0**	**0.00 %**

Figure 3.26: Historical Comparison

Gender representation allows them to view the rate of male to female students entering the university. A significant portion does not report this information.

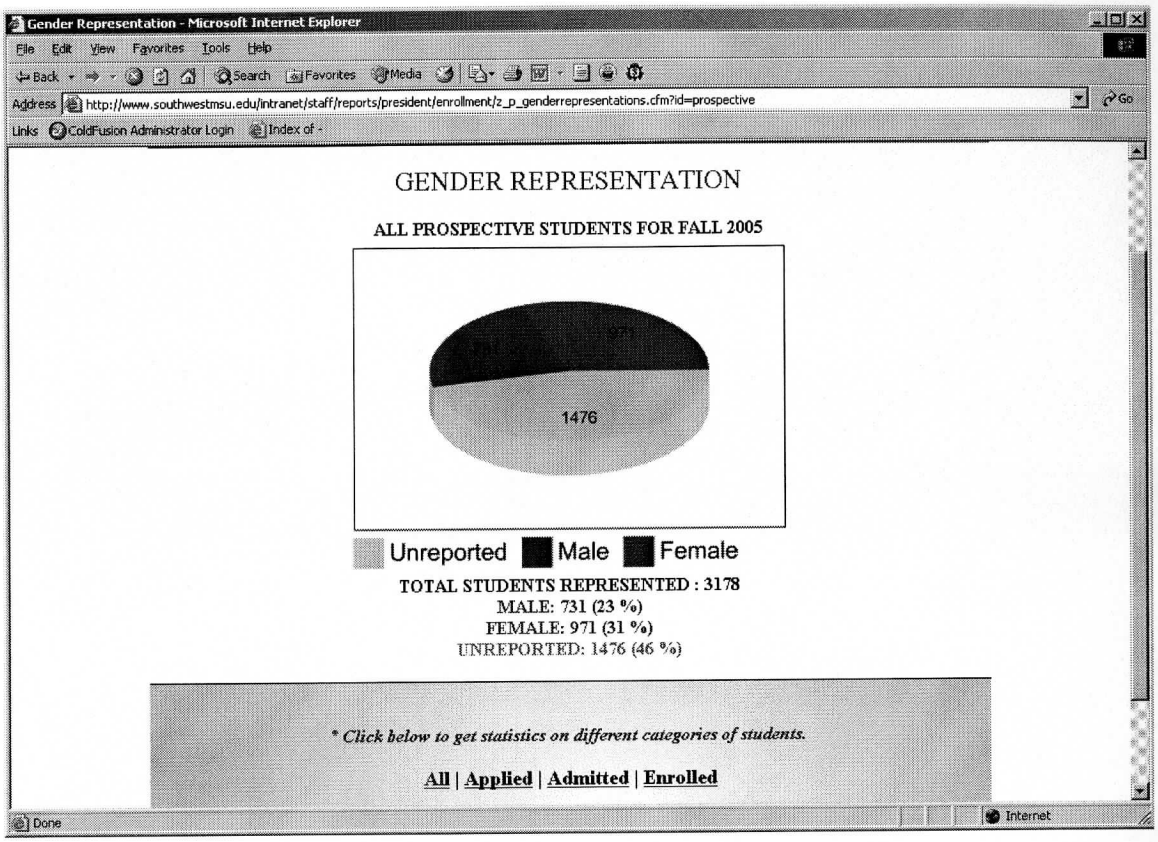


Figure 3.27: Gender Representation

Many scholarships and grants are based on the enrollment figures regarding ethnicity. By monitoring this statistic administration can view what percentage of incoming students will be enrolled in the university.

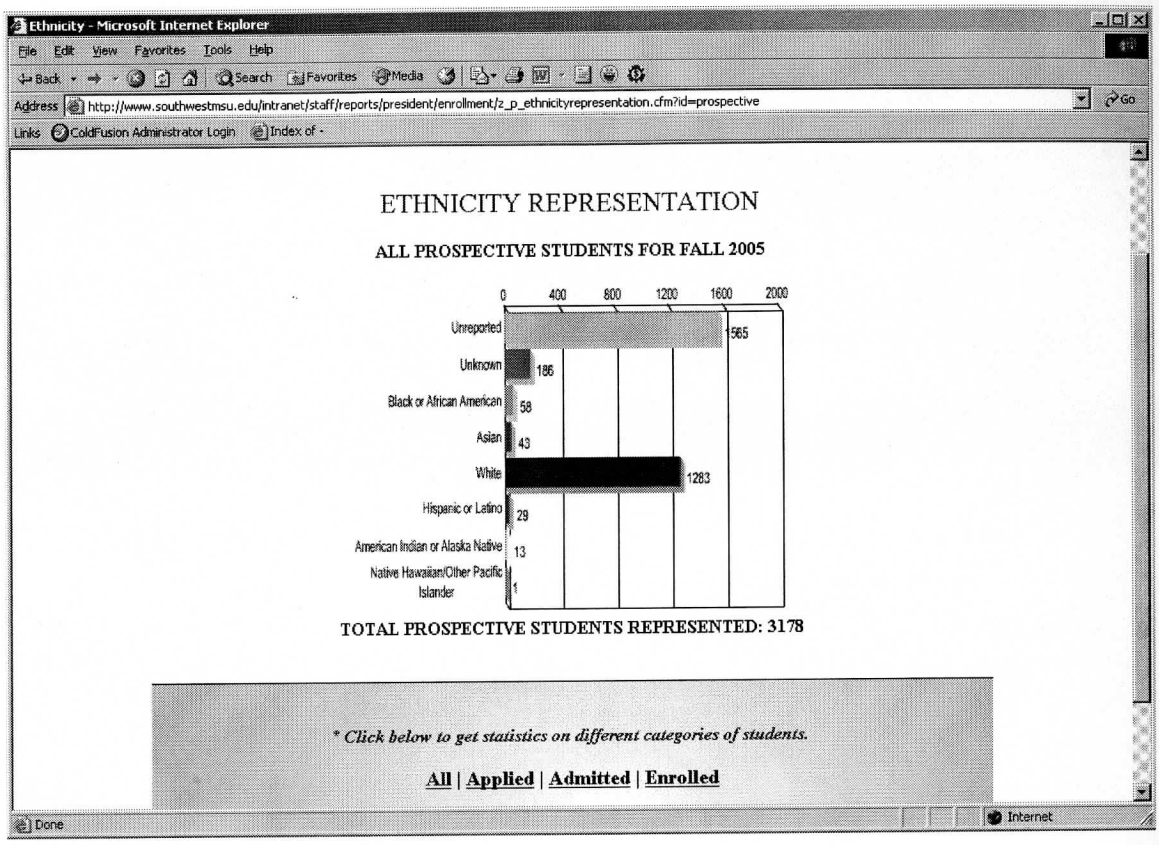


Figure 3.28: Ethnicity Representation

Enrollment tracks where each students home state. This allows enrollment to monitor progress of marketing outside of their current region.

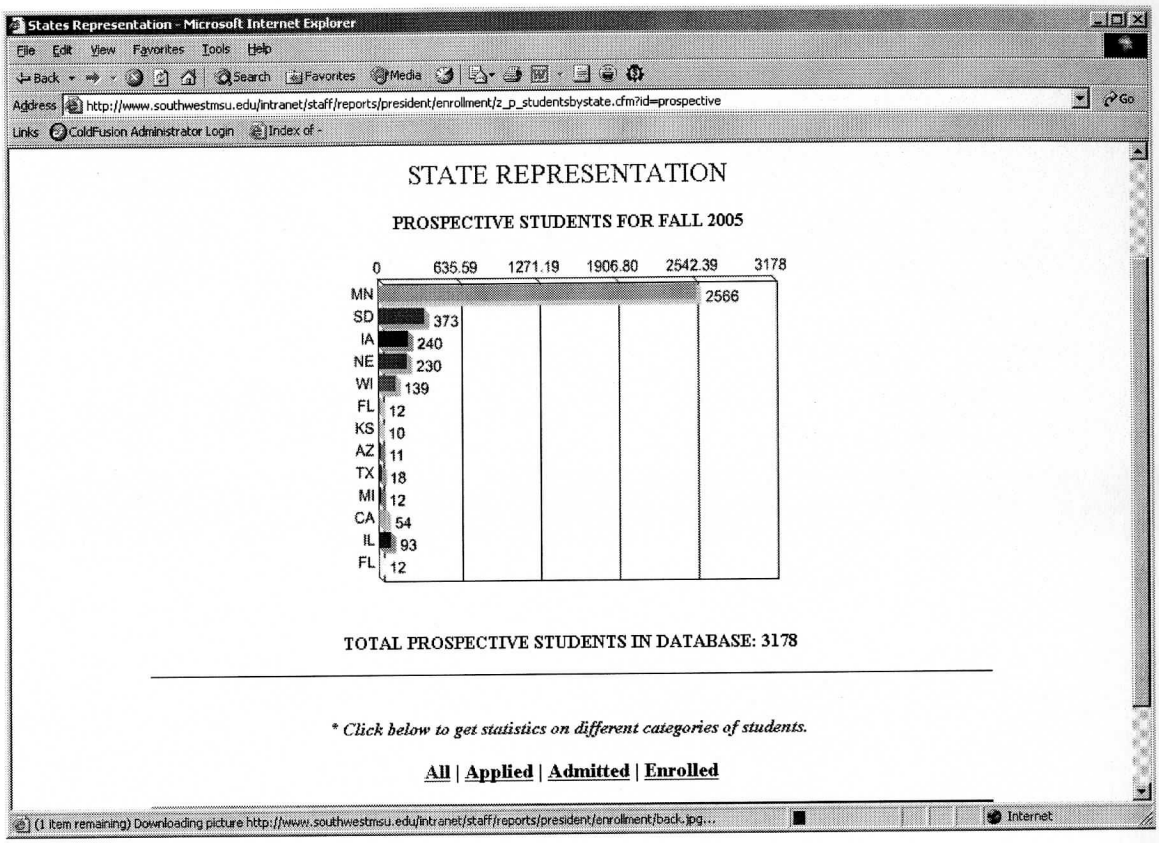


Figure 3.29: State Representation

Tracking majors can prepare the university for increases or decreases in selected majors which can help them to decide which programs will need additional support and which programs are being successful.

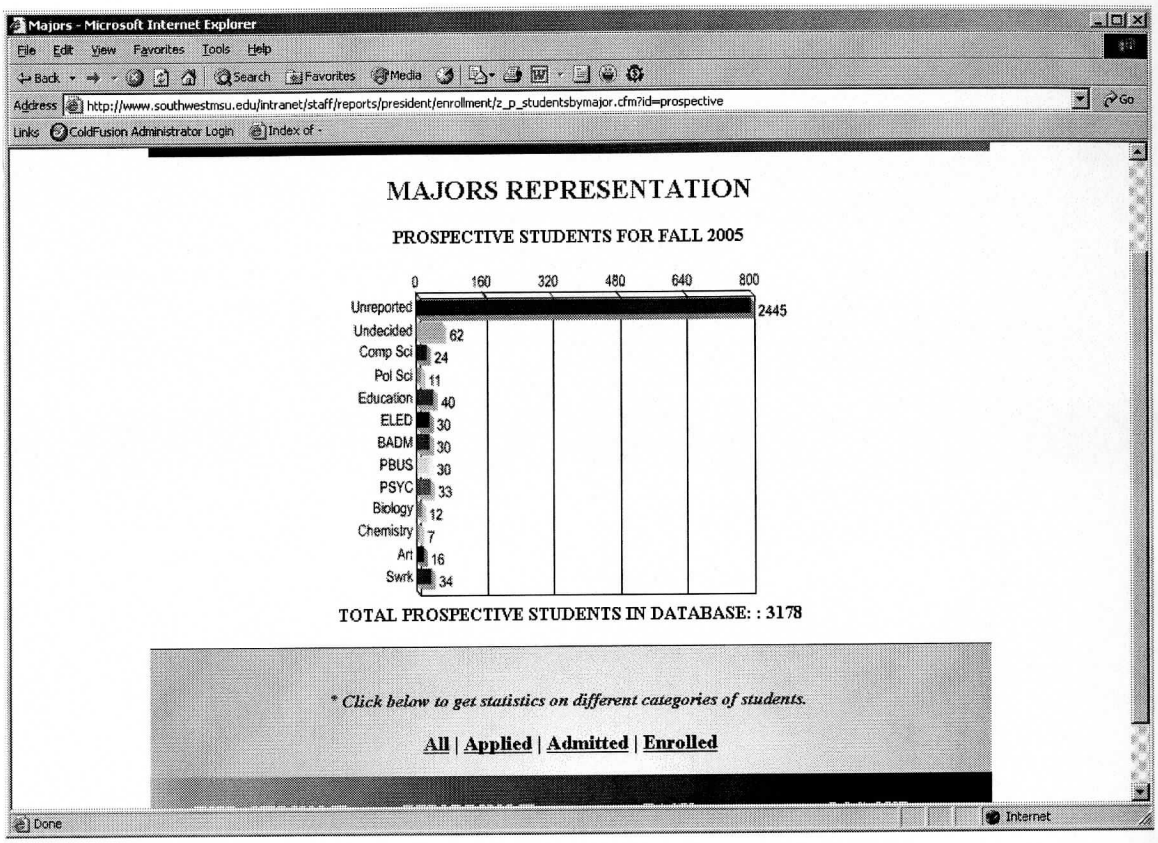


Figure 3.30: Major Representation

Tracking ACT and GPA scores immediately informs the administration as to the quality of the entering students as a whole. If this number decreases the quality of education can also decrease.

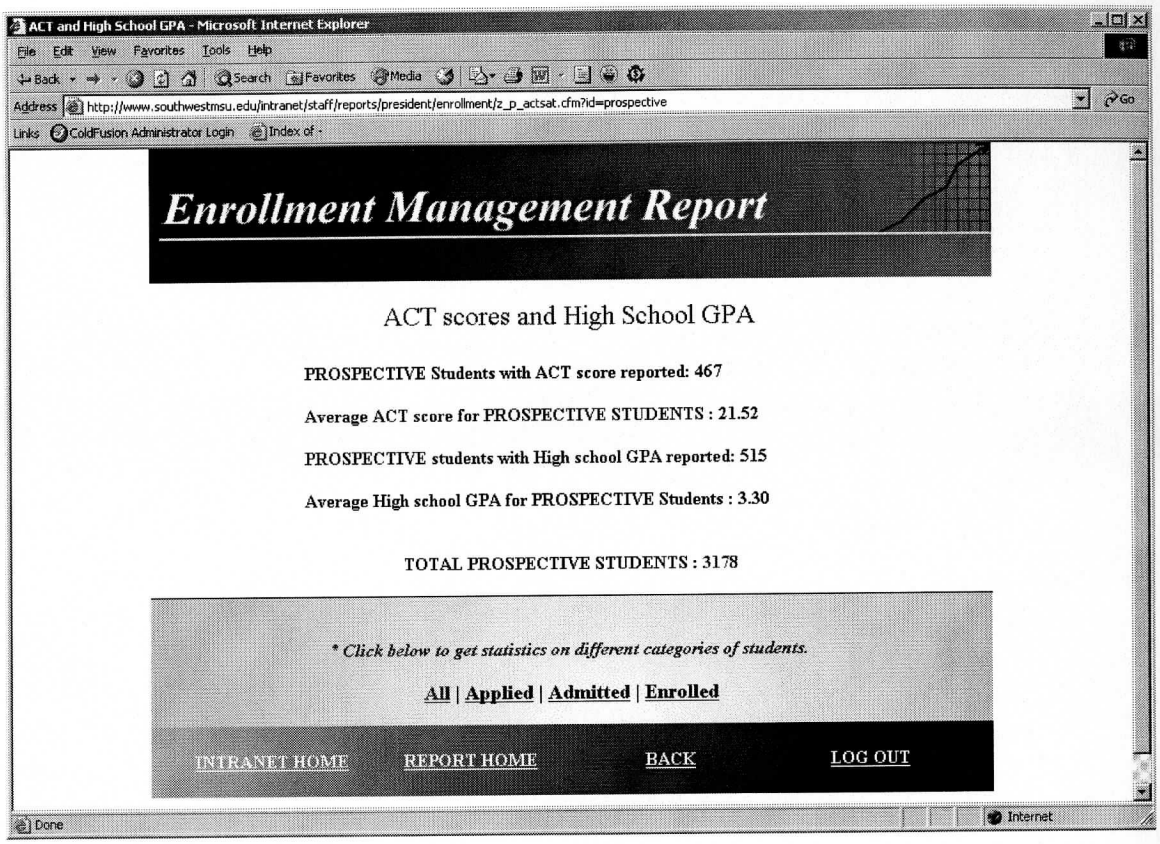


Figure 3.31: ACT and High School GPA Representation

Chapter 4 Results and Conclusions

The enrollment management systems level of success can be measured in multiple ways. As a time saving collaboration of information it has had an enormous impact on key university staff's daily duties. The centralized location of information presented in a report format or in a comma separated file has reduced workload by hours per week for multiple people. This time saved allows those people to focus on customer service and other areas of their job that can have impacts on enrollment numbers. The software also eliminates the need for printed material. The mailings sent by the enrollment office have dropped dramatically at a significant budget savings to the department. The money previously spent on producing individual mailings and postage has been redirected into other areas of emphasis. These two areas signify significant cost savings by having a software solution available to the campus.

Other revenue generation gained by the software is the ability to extend the enrollment offices reach. In prior years the university purchased names from the ACT organization in mostly a geographically close range. The reason this was done was to ensure the ability to travel and spend time at those particular locations. With the use of an electronic communications tool the offices reach has larger boundaries. The enrollment office is beginning plans to increase the spending on larger cities that are outside our traditional student pull areas. By purchasing names from states further away, such as Kansas, the office can market to more students at a lower cost giving them a competitive advantage over universities that can not reach as far.

Implemented throughout the first year, the enrollment management software has no historical data to model predictions of increased enrollment. The system has proven a number of factors that lead one to believe that increased enrollment may be a result of the product. A larger prospective student base, ability to target students with specific interests, and positive reactions from communications are indicators that the product has been successful in obtaining additional students. The enrollment management software is definitely a popular product for a department that is very new to this style of business process. The level of excitement has been quickly followed by suggestions of what we could do next. The enrollment office has transformed into an area of campus that is beginning to see the benefit to technology and how it can aid rather than hamper the enrollment process. The director of enrollment has been active in discussions regarding how the software can adapt and change to further enhance the prospective student experience.

4.1 IT Value and Justification

A personal study of the potential savings is key to showing how in house development projects save time, which translates to profits. In the following charts savings is represented by calculating an estimated timesavings per week multiplied by staff salaries over the course of a year. These savings represent the value of work time saved NOT actual dollar savings to the bottom line.

Table 4.1 Initial Costs

Initial Costs	Year 1	Year 2	Year 3	Year 4
Initial investment	$4,886	$0	$0	$0
Implementation	$0	$0	$0	$0
Ongoing support	$1,034	$1,034	$1,034	$1,034
Training	$0	$0	$0	$0
Other	$0	$0	$0	$0
Total costs	*$5,919*	*$1,034*	*$1,034*	*$1,034*

Table 4.2 Cash Flow Statement

Cash flow and ROI				
BENEFIT DRIVERS	**YEAR**			
	1	**2**	**3**	**4**
Labor Costs				
DBA Staff Position (ITS 3)	$12,373	$12,373	$12,373	$12,373
Admissions Secretary (2 staff members)	$12,278	$12,278	$12,278	$12,278
Admissions Counselors (6 counselors)	$55,394	$55,394	$55,394	$55,394
Admissions Administration (2 Administrators)	$15,446	$15,446	$15,446	$15,446
3rd Party Software Costs	$70,000	$5,000	$5,000	$5,000
Total annual benefits	$165,489.63	*$100,490*	*$100,490*	*$100,490*
Implementation filter	100%	100%	100%	100%
Total benefits realized	$165,489.63	*$100,490*	*$100,490*	*$100,490*

Table 4.3 Cumulative Savings

Savings	Year 1	Year 2	Year 3	Year 4
Annual Savings flow	*$159,570*	*$99,456*	*$99,456*	*$99,456*
Cumulative Savings flow	***$159,570***	***$259,027***	***$358,483***	***$457,939***

4.2 Future Plans for Enrollment Management Software

The enrollment management software has gone through a lot of changes since the initial meetings. The software has served the purpose that the enrollment office originally intended but interest has been expressed in adding both a section on a phone call scripting system as well as developing a more advanced statistical package to evaluate trends in students that enrolled versus those that did not. These additions would be seamlessly built into the existing structure to allow the software to become even more flexible.

The system designer and developer have learned several lessons associated to projects such as this. The ability to set down firm guidelines at the onset of the project would help to eliminate scope creep and meet expectations from the users of the system. Also the system developer has plans to provide a level of self-support in anticipation of customizing this software to be bundled and sold to other smaller universities that faced a similar budget problem that Southwest Minnesota State University faced.

References

Broadbent, Marianne., & McDonald, Mark., & Hunter, Richard. (June 2003). Does IT Matter? An HBR Debate. http://harvardbusinessonline.hbsp.harvard.edu/b02/en/files/misc/Web_Letters.pdf

Counselor Interviews. (2004). Meeting with Counselors.

Danahar, David, President (2004). Strategic Planning Day. From http://www.southwestmsu.edu/strategicplanning/

EMT Hobson's. (2004). Sales Presentation. http://www.emt.hobsons.com/

Gerber, Nancy (2004). Personal Interview.

Goral, Tim (March 2003). Intelligent Admissions – University Business [Electronic Version]. EdGenuiti.

Hoffer, Jeffery., & Prescott, Mary., & McFadden, Fred. (2002). Modern Database Management Seventh Edition. New Jersey

James Tower. (2005). Admissions Genie. http://www.jamestower.com/530.html

Macromedia. (2005). Coldfusion Server Whitepages. http://www.macromedia.com/software/coldfusion/productinfo/overview/fp_frameset.html

Pawlenty, Gov. (Jan. 2005). Minnesota FY 2006-2007 Budget Summary and Policy Highlights. Retrieved March, 27th, from http://www.budget.state.mn.us/budget/operating/200607/050125_summary.pdf

RightNow Technologies. (2005). Content Resources Management. http://www.rightnow.com/

Shearer, Richard (2004). Multiple Personal Interviews.

Steele, Jonathon. (2002). Understanding College Bound Students and Communicating With Them Effectively. Journal of College Admission http://www.findarticles.com/p/articles/mi_qa3955/is_200204/ai_n9036935/pg_2

Thompson, RaNae. (2004). MnScu Project Manager Interview.

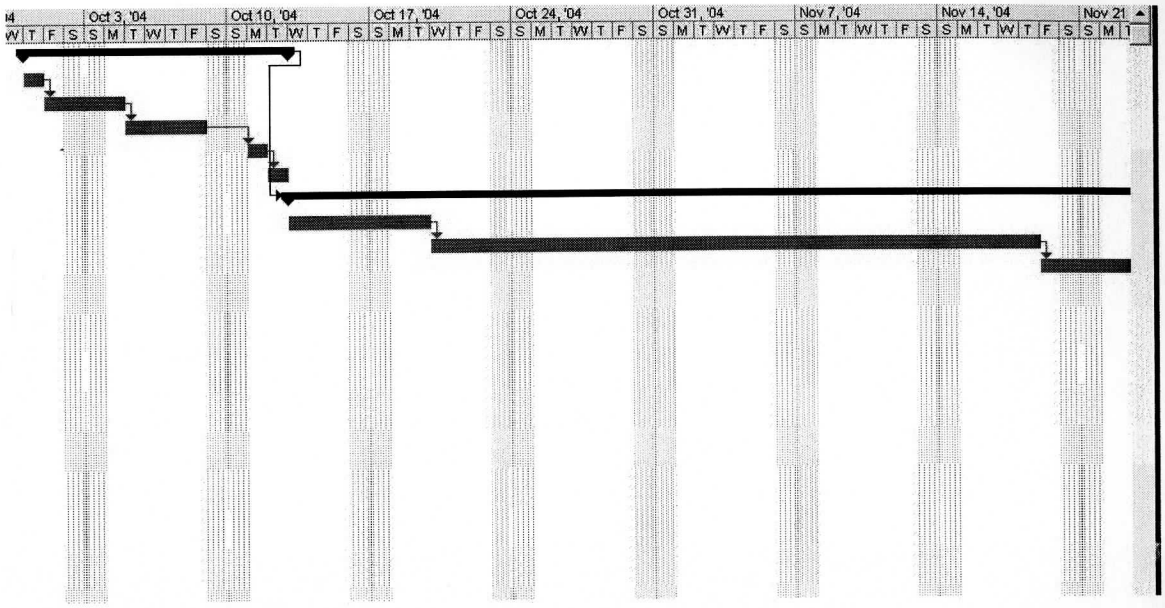
Webopedia. Batch File. Retrieved May 1, From http://www.webopedia.com/TERM/b/batch_file.html

Appendix A

Table 4.4: Work Breakdown Structure

	WBS	Task Name	Leveling Delay	Duration	Start	Finish	Successors
1	**1**	**Database**	**0 edays**	**9 days**	**Thu 9/30/04**	**Tue 10/12/04**	**7**
2	1.1	Identify ISRS tables	0 edays	1 day	Thu 9/30/04	Thu 9/30/04	3
3	1.2	Create ODBC Linked T	0 edays	2 days	Fri 10/1/04	Mon 10/4/04	4
4	1.3	Create Local Tables	0 edays	4 days	Tue 10/5/04	Fri 10/8/04	5
5	1.4	Download Data	0 edays	1 day	Mon 10/11/04	Mon 10/11/04	6
6	1.5	Created Batch File Aut	0 edays	1 day	Tue 10/12/04	Tue 10/12/04	
7	**2**	**Core Query Function**	**0 edays**	**32 days**	**Wed 10/13/04**	**Thu 11/25/04**	**11**
8	2.1	Create Set Query	0 edays	5 days	Wed 10/13/04	Tue 10/19/04	9
9	2.2	Create Build Query	0 edays	22 days	Wed 10/20/04	Thu 11/18/04	10
10	2.3	Develop "And" "Or" fur	0 edays	5 days	Fri 11/19/04	Thu 11/25/04	
11	**3**	**Core Email Function**	**0 edays**	**8 days**	**Fri 11/26/04**	**Tue 12/7/04**	**16**
12	3.1	Create Email Add on's	0 edays	3 days	Fri 11/26/04	Tue 11/30/04	14,15
13	3.2	Create Email Content W	0 edays	1 day	Fri 11/26/04	Fri 11/26/04	14,15
14	3.3	Create Email Report pa	0 edays	2 days	Wed 12/1/04	Thu 12/2/04	15
15	3.4	Create Export Pages	0 edays	3 days	Fri 12/3/04	Tue 12/7/04	
16	**4**	**Reports**	**0 edays**	**14.5 days**	**Wed 12/8/04**	**Tue 12/28/04**	**30**
17	4.1	Counselor Email Repor	0 edays	2 days	Wed 12/8/04	Thu 12/9/04	24
18	**4.2**	**Counselor Territory**	**0 edays**	**0.5 days**	**Wed 12/8/04**	**Wed 12/8/04**	**21,22,24**
19	4.2.1	Counselor Territory	0 edays	0.5 days	Wed 12/8/04	Wed 12/8/04	
20	4.2.2	Counselor Territory	0 edays	0.5 days	Wed 12/8/0	Wed 12/8/04	
21	4.3	Counselor Progress Cl	0 edays	1 day	Wed 12/8/04	Thu 12/9/04	24
22	**4.4**	**Enrollment Director**	**0 edays**	**2 days**	**Wed 12/8/04**	**Fri 12/10/04**	**24**
23	4.4.1	All Counselor Activ	0 edays	2 days	Wed 12/8/04	Fri 12/10/04	
24	**4.5**	**Prospective Report**	**0 edays**	**2 days**	**Fri 12/10/04**	**Tue 12/14/04**	**29**
25	4.5.1	Gender	0 edays	2 days	Fri 12/10/04	Tue 12/14/04	
26	4.5.2	ACT	0 edays	2 days	Fri 12/10/04	Tue 12/14/04	
27	4.5.3	GPA	0 edays	2 days	Fri 12/10/04	Tue 12/14/04	
28	4.5.4	Ethnicity	0 edays	2 days	Fri 12/10/04	Tue 12/14/04	
29	4.6	Administrative Enrollme	0 edays	10 days	Tue 12/14/04	Tue 12/28/04	
30	**5**	**Return On Investment S**	**0 edays**	**2 days**	**Tue 12/28/04**	**Thu 12/30/04**	
31	5.1	Collect Salary Info	0 edays	2 days	Tue 12/28/04	Thu 12/30/04	
32	5.2	Estimate Time Saved	0 edays	1 day	Tue 12/28/04	Wed 12/29/04	

Figure 4.1: Gantt Chart



Appendix B

Coldfusion Query Code for Selector Page

```
<CFIF IsDefined("session.tech_id") IS "No">
 <cflocation url="/intranet/staff/login.cfm" addtoken="No">
</CFIF>
<cfset olddate = #DateAdd('D', -7, Now())#>
<cfset olddate2 = #DateFormat(olddate, "YYYYMMDD")#>
<cfset current = #DateFormat(now(), "YYYYMMDD")#>
<cfquery datasource="EMT" name="majors" cachedwithin="#CreateTimeSpan(0,1,0,0)#"
blockfactor="100">
select distinct major from emt_prospective_students
</cfquery>
<cfquery datasource="EMT" name="yearterm" cachedwithin="#CreateTimeSpan(0,1,0,0)#"
blockfactor="100">
SELECT DISTINCT EMT_Prospective_Students.YRTR,
EMT_Valid_Year_term.val_yrtr_longnm
FROM EMT_Prospective_Students, EMT_Valid_Year_term WHERE
EMT_Prospective_Students.YRTR
= EMT_Valid_Year_term.Val_Yrtr
</cfquery>
<cfquery datasource="EMT" name="counselors_ter" cachedwithin="#CreateTimeSpan(0,1,0,0)#"
blockfactor="100">
select * from emt_counselor_email
</cfquery>
<cfquery datasource="EMT" name="actscores" cachedwithin="#CreateTimeSpan(0,1,0,0)#"
blockfactor="100">
select DISTINCT actscore from emt_prospective_students
</cfquery>
```

```
<cfquery datasource="EMT" name="counselors" cachedwithin="#CreateTimeSpan(0,1,0,0)#"
blockfactor="100">
select DISTINCT counselor_name from emt_prospective_students
</cfquery>
<cfquery datasource="EMT" name="highschools" cachedwithin="#CreateTimeSpan(0,1,0,0)#"
blockfactor="100">
SELECT distinct EMT_Prospective_Students.HSInstId, EMT_Valid_HighSchools.NP_NAME,
EMT_Valid_HighSchools.CITY FROM {oj EMT_Valid_HighSchools RIGHT OUTER JOIN
EMT_Prospective_Students
ON EMT_Valid_HighSchools.HSInstCode = EMT_Prospective_Students.HSInstId } order
by EMT_Valid_HighSchools.NP_NAME
</cfquery>
<cfquery datasource="EMT" name="cities" cachedwithin="#CreateTimeSpan(0,1,0,0)#"
blockfactor="100">
select distinct city from emt_prospective_students
</cfquery>
<cfquery datasource="EMT" name="contacts" cachedwithin="#CreateTimeSpan(0,1,0,0)#"
blockfactor="100">
select distinct contact,contactshortnm from emt_valid_contact_codes
</cfquery>
<cfquery datasource="EMT" name="contact_date" cachedwithin="#CreateTimeSpan(0,1,0,0)#"
blockfactor="100">
select distinct cntct_date from emt_contacts
</cfquery>
```

Appendix C

Emt_Prospective_Students

- Tech_ID
    - Used as a unique identifier in all databases Primary Key
- YRTR
    - Denotes the year in which the student plans on entering a university. For example if you have a year term of 20063 you plan on entering a university in the Fall of 2005 whereas a year term of 20065 would be the Spring 06 session.
- Last_Name
    - Last Name of the student
- First_Name
    - First Name of the student
- Middle_Name
    - Middle Name of the student
- Birth_Date
    - The date on which the student was born
- Gender
    - Denotes male, female or unspecified
- Origin_cnty
    - County in which the student resides
- Origin_State
    - State in which the student resides

- Ethnic_code
    - Ethnic Status
- Stop_Cntct_Code
    - Denotes someone who no longer wants to be contacted.
- Internet_addr
    - Email address
- Street
    - Home address
- City
    - Home Town
- State
    - State in which the student lives
- ZIP
    - ZipCode
- Tel_nbr
    - Telephone Number
- AppliedStat
    - If the student has applied it is represented in this field
- AppliedDate
    - Date the student applied
- ADMSTAT
    - State of the admission process for a particular student
- ADMDATE

- Date of the admission process for a particular student

- ENRCAT
  - Enrollment category
- HSInstID
  - High school the student attends
- HSGPA
  - High School Grade Point Average
- Major
  - Degree student is pursuing
- ACTScore
  - Score of ACT

EMT_Contacts

- Tech_ID
  - Primary Key, unique identifier for student
- Contact_yrtr
  - Year term in which the student was contacted
- Cntct_seq_nbr
  - Contact order
- Cntct_src_code
  - Unique contact identifier
- ContactShortNm
  - Contact title in short form
- Cntct_date

- Date of the contact

- ADD_User_Id
  - Person who entered the contact
- Comment
  - Text message from author of contact listing

EMT_EMAIL_STATS

- ID
  - Autonumber, Primary Key
- Emailed
  - Unique number given to each email batch sent
- Counselor_Name
  - Name of the counselor sending the batch email
- No_Of_Students
  - Total number of students the email was sent to
- Query_Used
  - A record of which criteria the counselor chose
- Date_time
  - A record of when the email was sent
- Subject
  - Subject of the email
- Content
  - The body of the email message